

THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES

Établissement d'horaires d'examens par coloration de graphes

Ferauge, Jacques

Award date:
1983

Awarding institution:
Université de Namur

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Facultés
universitaires
N.D. De la paix
Namur

Institut d'informatique

Etablissement d'horaires
d'examens par coloration
de graphes.

Jacques FERAUGE

Mémoire présenté en vue de
l'obtention du grade de
licencié en informatique.

Année Académique 1982-1983

Je remercie vivement Monsieur Jean FICHEFET d'avoir accepté la direction de ce mémoire

Je tiens à exprimer toute ma reconnaissance à Jean-Paul LECLERCQ pour l'aide efficace apportée tout au long de ce travail. Il m'a donné entière liberté de faire un travail personnel tout en m'aidant par ses conseils et remarques. Je lui suis surtout très reconnaissant pour l'exprit ouvert et l'ambiance sympathique qui m'on accompagnés pendant la réalisation de ce mémoire.

Je remercie finalement tous ceux qui de près ou de loin ont contribué à la réalisation de ce travail.

TABLE DES MATIERES

Introduction

Chapitre I. L'horaire d'examen comme modèle de graphe

- I.1. Définition du problème et mise en modèle
 - I.1.1. Position du problème
 - I.1.2. Un modèle de graphe
- I.2. Mise en modèle du problème
 - I.2.1. Détermination et identification des séances d'examen d'un cours
 - I.2.2. Détermination des composants d'un sommet
 - I.2.2.1. Attribution du cours
 - I.2.2.2. Attribution du professeur
 - I.2.2.3. Attribution des étudiants
- I.3. Première approche du processus de coloration du graphe

Chapitre II. L'horaire d'examen complet comme modèle de graphe avec contraintes de temps

- II.1. Le modèle de graphe et les contraintes directes
 - II.1.1. Les contraintes directes
 - II.1.2. Graphe avec fonction de temps
- II.2. Algorithme de coloration exploitant la matrice de temps
 - II.2.1. Choix du sommet à colorier
 - II.2.2. Choix de la couleur
 - II.2.3. Retour en arrière
- II.3. Les contraintes: structure des fichiers et implémentation
 - II.3.1. Les classes
 - II.3.2. Les étudiants
 - II.3.3. Les professeurs
 - II.3.4. Les cours
- II.4. Coloration d'un sommet i par une couleur k
 - II.4.1. Sommets ayant même professeur
 - II.4.2. Sommets ayant même étudiant
 - II.4.3. Sommets ayant même cours
- II.5. Conclusion
- II.6. L'implémentation

Chapitre III. Le programme d'horaire d'examen

- III.1. Gestion de fichier et saisie de données
 - III.1.1. Module de gestion des fichiers
 - III.1.1.1. Gestion du fichier faculté
 - III.1.1.2. Gestion du fichier cycle
 - III.1.1.3. Gestion du fichier date
 - III.1.1.4. Gestion du fichier classe
 - III.1.1.5. Gestion du fichier étudiant
 - III.1.1.6. Gestion du fichier professeur
 - III.1.1.7. Gestion du fichier cours
 - III.1.2. Modules intermédiaires

- III.1.2.1. Identification d'un professeur
- III.1.2.2. Identification d'un étudiant
- III.1.2.3. Identification d'une classe
- III.1.2.4. Modification d'un cours
- III.1.2.5. Saisie d'un cours
- III.1.3. Modules application
 - III.1.3.1. Module saisie des cours
 - III.1.3.2. Module saisie des étudiants
 - III.1.3.3. Module saisie des professeurs
 - III.1.3.4. Module saisie des modalités d'examen
 - III.1.3.5. Module inscription aux examens
 - III.1.3.6. Module indisponibilité d'un professeur
 - III.1.3.7. Module date de la session
- III.1.4. Modules menus
- III.2. Calcul de l'horaire

Chapitre IV. Expériences pratiques

- IV.1. Tests effectués
- IV.2. Quelques résultats

Conclusion

Bibliographie

I N T R O D U C T I O N

La conception d'un horaire d'examens d'un établissement scolaire est depuis longtemps un casse-tête délicat et compliqué. Chaque année, ce travail très lourd préoccupe les responsables administratifs de nombreuses écoles, mais de bons horaires sont rarement trouvés. L'apparition de l'informatique comme outil de gestion a fait naître le désir d'utiliser l'ordinateur comme soutien lors de l'établissement d'un horaire.

Les horaires conçus manuellement présentent des inconvénients dûs entre autres au fait que l'ampleur du problème combinatoire dépasse les capacités d'appréhension globale d'un individu. Il en résulte des oublis, des incompatibilités, des difficultés de modification, une dépense d'énergie et de temps considérables ainsi que des difficultés liées à un manque d'information (modalité d'examens, indisponibilités,...)

De nombreuses conversations avec les responsables des horaires ont fourni une vue globale et assez complète du problème. A côté des données de base (étudiants, classes, cours, professeurs), on distingue de nombreuses contraintes (ne pas placer deux examens différents d'un même étudiant à la même période, indisponibilités de professeurs, consécutivité des oraux,...) de nature pratique ou pédagogique.

Le but de ce travail est dès lors d'établir un logiciel résolvant tous les problèmes rencontrés et qui soit utilisable par des non-spécialistes en informatique. Ce programme doit donc gérer les informations nécessaires, effectuer les calculs proprement dits et sortir sous forme compacte les résultats obtenus.

Pour réaliser un outil de gestion répondant aux exigences réelles, il faut surtout trouver un modèle mathématique approprié. Nous travaillons avec un modèle de graphe qu'on va adapter aux contraintes envisagées, et nous utilisons un algorithme dit de "coloration de graphes". Le premier chapitre décrit le modèle mathématique et la mise en modèle du problème. Ce modèle sera modifié au deuxième chapitre afin d'inclure les contraintes rencontrées en pratique; nous donnons en plus les grands principes de l'algorithme de coloration utilisé et la structure des données et contraintes relatives à l'horaire. Le troisième chapitre analysera la structure et les différentes parties du programme. (saisie de données et gestion

des fichiers, calcul de l'horaire).
Une série de résultats pratiques seront fournis au quatrième chapitre.

C H A P I T R E I

Chapitre I : L'horaire d'examen comme modèle de graphe

=====

I.1. Définition du problème et mise en modèle

=====

Le problème consiste à calculer l'horaire d'examen d'un établissement scolaire quelconque. Etant donné un ensemble de cours, il s'agit de trouver des périodes de session (= par exemple : une demi-journée) pour chaque épreuve (orale ou écrite) de cours tout en respectant les contraintes évidentes du type "aucun étudiant ne peut avoir deux examens différents pendant la même période". Donc, il faut s'arranger pour qu'aucun étudiant et aucun professeur ne soit pris par plus d'un examen à la même période. Ce sont les contraintes indirectes du problème. Ce premier chapitre décrit un modèle approprié qui sera modifié au deuxième chapitre en ajoutant d'autres contraintes de nature pédagogique (dites contraintes directes).

I.1.1. Position du problème

=====

Soient données au départ les ensembles suivants :

- les étudiants $\mathcal{E} = \{e_i \text{ tq } i \in I_1\}$
- les professeurs $\mathcal{P} = \{p_j \text{ tq } j \in I_2\}$
- les cours $\mathcal{C} = \{c_k \text{ tq } k \in I_3\}$

Un examen à donner se présente comme un sous-ensemble de $\mathcal{E} \cup \mathcal{P} \cup \mathcal{C}$
 c-à-d. qu'un examen fait intervenir pendant une période un ensemble d'étudiants, de professeurs et porte sur un cours.

Exemple : l'examen fait intervenir pendant une demi-journée les

étudiants e_1, e_2, e_3 , le professeur p_1 et porte sur le cours c_1 . Il se présente sous la forme $\{e_1, e_2, e_3, p_1, c_1\}$

Soit X l'ensemble des scéances d'examens à donner et x un élément quelconque de X . Donc, $X \subseteq \{x \text{ tq } x \text{ est sous-ensemble de } \{U, P, V, E\}\}$
 Le calcul de l'horaire d'examen revient donc à donner une période à chaque examen x .

I.1.2. Un modèle de graphe

=====

Soit donné un ensemble $E = \{U, P, V, E\}$
 $X \subseteq \{x \text{ tq } x \text{ sous-ensemble de } E\}$ ou X est l'ensemble des sommets

$U = \{(x, y) \text{ tq } x, y \in X, x \neq y, x \text{ } y \neq \emptyset\}$
 où U est l'ensemble des arcs

$H = \{h \text{ tq } h \in H\}$ l'ensemble des couleurs.

A une scéance d'examen correspond un sommet du graphe. Deux sommets du graphe sont liés par un arc (on dit qu'ils sont adjacents) si les deux scéances correspondantes possèdent au moins un élément (étudiant, professeur, cours) commun. Dans la suite, on raisonnera sur ce graphe $G(X, U)$. Le calcul d'horaire d'examen se ramène ainsi à donner une couleur à chaque sommet du graphe (donner une demi-journée à chaque scéance) tout en respectant la contrainte que deux sommets adjacents (deux scéances possédant un élément en commun) ne puissent pas prendre la même couleur.

A chaque couleur correspond une demi-journée.

Finalement, on est ramené à un problème de coloration de graphe.

colorer les sommets de façon à ce que deux sommets adjacents ne prennent pas la même couleur.

Le fait que le sommet x prend la couleur h se note $f(x) = h$: l'application "coloration" associe à chaque sommet une couleur $f : X \rightarrow H$ $f(x)$.

I.2. Mise en modèle du problème

=====

Ce paragraphe traite de l'application qui, à un cours donné, fait correspondre un certain nombre de séances d'examen, c-à-d., si l'on parle en terme du modèle de graphe exposé dans le paragraphe précédent, à un certain nombre des sommets du graphe tout en déterminant les composants (cours, professeur, étudiants) de ce sommet.

La première partie de ce paragraphe traitera de cette application en terme "quantitatif", c-à-d. détermination du nombre de sommets et identification des cours correspondant à un sommet.

La deuxième partie de ce paragraphe traitera de cette application en terme de "qualité", c-à-d. la détermination des composants d'un sommet.

I.2.1. Détermination et identification des séances d'examen d'un cours

=====

L'expérience quotidienne nous apprend qu'un cours donné durant l'année scolaire se transforme durant la session d'examen, en une épreuve écrite et/ou une épreuve orale.

L'épreuve écrite étant la même pour tout le monde, elle consiste en une seule séance d'une demi ou d'une journée.

L'épreuve orale, elle, consiste le plus souvent pour le professeur, en plusieurs demi-journées d'interrogation. Dans tout ce qui suit, on appellera séance d'examen : une demi-journée consacrée à une épreuve écrite ou orale portant sur un cours, ou autrement dit une séance est un sommet du graphe.

Le problème consiste donc à déterminer une application f

$$f : \mathcal{C} \longrightarrow X : c \longrightarrow x$$

- A un élément $c \in \mathcal{C}$ peut correspondre 0, 1 ou plusieurs $x \in X$.
- A un élément $x \in X$ ne peut correspondre qu'un seul $c \in \mathcal{C}$.

En attribuant un n° logique i à chaque $c \in C$

un n° logique k à chaque $x \in X$

Ou autrement dit, si l'on introduit 2 fonctions g et g'

$$g : C \longrightarrow N : c \longrightarrow i$$

$$\text{tq } c \neq c' \Rightarrow g(c) \neq g(c')$$

$$g' : X \longrightarrow N : x \longrightarrow k$$

$$\text{tq } x \neq x' \Rightarrow g'(x) \neq g'(x')$$

Alors on peut considérer l'application f équivalente à une application f'

$$f' : N \longrightarrow N : i \longrightarrow k$$

Si de plus, on fait l'hypothèse qu'à un entier positif i correspondant par g à un $c \in C$ ne peuvent correspondre par f' que des entiers positifs k consécutifs ($k; k+1, \dots, k+n-1$) correspondant par g' à des sommets $x \in X$, alors ces suites d'entiers positifs consécutifs peuvent être déterminés par 2 entiers

1) k : le plus petit entier image de i par f'

2) n : le nombre d'entiers image de i par f'

C'est ainsi que l'on peut également considérer l'application f comme équivalente à une application

$$f'' : N \longrightarrow N \times N : i \longrightarrow (j, k)$$

$$\text{tel que si } f(i) = (j, k) \quad j, k \neq 0$$

$$\text{alors } \forall i' (i \neq i') \text{ tel que } f(i') = (j', k') \text{ et } j', k' \neq 0$$

$$j' + k' < j \text{ ou } j' > j + k - 1$$

Exemple : $f''(4) = (5, 3)$

Conséquence : les sommets n° 5, 6, 7 correspondent au cours n° 4.

Conséquence : un sommet n° i ($i > 0$) ne correspond qu'à au plus un seul cours

Dans la pratique de l'horaire d'examen, il sera utilisé une fonction g

$$g : N \longrightarrow N \times N \times N : i \longrightarrow (j, k, l)$$

tel que si $g(i) = (j, k, l)$ alors

- si $j \neq 0$ alors le sommet n° j correspond à l'épreuve écrite portant sur le cours n° i .

sinon, l'examen portant sur le cours n° i ne comporte pas d'épreuve écrite.

- si $k \neq 0$ alors les sommets n° $k, k+1, \dots, k+l-1$, correspondent aux séances orales d'examens portant sur le cours n° i

sinon l'examen portant sur le cours n° i ne comporte pas d'épreuve orale.

Propriétés de la fonction g

=====

- ```
1) si $g(i) = (j, k, l)$ $j, k \neq 0$
 alors $k = j + 1$
```

Cette propriété assure que les sommets correspondants à des séances portant sur un même cours sont consécutifs.

- 2) si  $g(i) = (j, k, l)$  alors  $k = 0 \iff l = 0$

Cette propriété assure que si un sommet correspond à une séance d'épreuve orale d'un cours, ce cours comporte au moins une épreuve orale et inversement.

- 3) si  $g(i) = (j, k, l)$   $j \neq 0$  ou  $k \neq 0$   
 alors en posant  $s_{\min} = j$  si  $j \neq 0$   
 $= k$  sinon  
 $s_{\max} = k + l - 1$  si  $k \neq 0$   
 $= j$  sinon  
 $\forall i' \text{ t.q. } g(i') = (j', k', l')$   $j' \neq 0$  ou  $k' \neq 0$   
 en posant  $s_{\min}' = j'$  si  $j' \neq 0$   
 $= k'$  sinon  
 $s_{\max}' = k' + l' - 1$  si  $k' \neq 0$   
 $= j'$  sinon  
 $s_{\min}' > s_{\max}$  ou  $s_{\max}' < s_{\min}$

Cette propriété assure que si un sommet correspond à un cours, il ne correspond qu'à ce seul cours



4) si  $g(i) = (j, k, l)$   $j \neq 0$  ou  $k \neq 0$   
 en posant  $s_{\min} = j$  si  $j \neq 0$   
 $= k$  sinon  
 $s_{\max} = k + l - 1$  si  $k \neq 0$   
 $= j$  sinon  
 alors  $\forall s_{\min} \leq k \leq s_{\max}$ , le sommet n°  $k$  correspond à une  
 séance du cours n°  $i$   
 $\forall 1 \leq k \leq s_{\min}, \exists i' < i$  tel que  $g(i') = (j', k', l')$   
 $j' \neq 0$  ou  $k' \neq 0$  tel que en posant  $s_{\min}' = j'$  si  $j' \neq 0$   
 $= k'$  sinon  
 $s_{\max}' = k' + l' - 1$  si  $k' \neq 0$   
 $= j'$  sinon  
 $s_{\min}' \leq k \leq s_{\max}'$

Cette propriété assure que si un sommet n°  $k$  correspond à un cours n°  $i$ ,  
 alors  $\forall k' \leq k, \exists i' \leq i$  t.q. le sommet n°  $k'$  correspond au cours n°  $i'$ .

Globalement, ces quatres propriétés nous assurent que

en posant nb-cours tel que 1)  $g(\text{nb-cours}) = (j, k, l)$   $j$  ou  $k \neq 0$

2)  $\forall i > \text{nb-cours}$   $g(i) = (0, 0, 0)$

nb-scea tel que 1) le sommet n° nb-scea correspond au  
 cours n° nb-cours

2)  $\forall s > \text{nb-scea}$ , le sommet n°  $s$  ne corres-  
 pond à aucun cours.

alors  $\forall 1 \leq s \leq \text{nb-scea}, \exists ! i$  tel que : 1)  $g(i) = (j, k, l)$   
 et en posant  $s_{\min} = j$  si  $j \neq 0$   
 $= k$  sinon  
 $s_{\max} = k + l - 1$  si  $k \neq 0$   
 $= j$  sinon  
 2)  $s_{\min} \leq s \leq s_{\max}$

En pratique, cela signifie qu'il est possible de retrouver, à partir  
 d'un cours, tous ses sommets correspondants, et à partir d'un sommet son  
 cours correspondant.

# Implémentation de la fonction g

=====

Si  $g(i) = (j,k,1)$  alors  $c-c(i,1) = j$   $c-c(i,2) = k$   $c-c(i,3) = 1$

En implémentant la fonction g, on va de plus assurer

- 1) pour chaque cours n° i  $1 \leq i \leq \text{nb-cours}$ , la correspondance avec les sommets du graphe X
- 2) le calcul de nb-scea, nombre de sommets du graphe
- 3)  $\forall$  sommet n° i  $1 \leq i \leq \text{nb-scea}$ , le nombre d'étudiants devant le composer  
 $= \text{nb-et-sc}(i)$

## Procédure const-scéance

=====

- 1) nb-scea = 0
- 2) Pour i = 1 jusque nb-cours faire
  - si le cours n° i a un écrit :  $\text{nb-scea} = \text{nb-scea} + 1$   
 $c-c(i,1) = \text{nb-scea}$   
 $\text{nb-et-sc}(\text{nb-scea}) = \text{nb-ins}(i)$
  - sinon  $c-c(i,1) = 0$
  - si le cours n° i a un oral : appeler nbreorale (i)  
 $c-c(i,2) = \text{nb-scea} + 1$   
 $c-c(i,3) = n$   
 $\text{nb-scea} = \text{nb-scea} + n$
  - sinon  $c-c(i,2) = 0$   $c-c(i,3) = 0$

3) retour

La procédure nbreorale (i) va calculer pour le cours n° i, le nombre de scéances d'orale et pour chaque scéance, le nombre d'étudiants qui doivent la composer.

- 1) Le nombre de scéances d'orales peut être déterminée de 2 façons
  - soit qu'il est fixé au départ : pour chaque cours, le nombre de scéances d'orale peut être considéré comme un paramètre en entrée  
 $= \text{nb-sc}(i)$        $n = \text{nb-sc}(i)$



- soit qu'il est calculé à partir de deux paramètres en entrée
  - 1) nb-et(i) = nombre d'étudiants qu'un professeur est disposé à interroger en une demi-journée
  - 2) nb-ins(i) = nombre d'étudiants inscrits à l'examen portant sur le cours i
 dans ce cas  $n = \text{nb-ins}(i) \text{ div nb-et}(i)$   
 si  $\text{nb-ins}(i) \bmod \text{nb-et}(i) \neq 0$  alors  $n = m + 1$

#### Exemples

1) nb-sc(i) = 3    alors  $n = 3$

2) nb-sc(i) = 0

nb-et(i) = 7

Nb-ins(i) = 54

nb-sc(i) = 0    2ème méthode de calcul du nombre de scéances orales

d'où  $n = 54 \text{ div } 7 = 7$

$54 \bmod 7 = 5 \neq 0$  d'où  $n = m + 1 = 8$

Remarques : lorsqu'un cours de n° i consiste notamment en une épreuve orale, il est impossible d'avoir en même temps nb-sc(i) = 0 et nb-et(i) = 0 car les valeurs obtenues de l'opérateur sont contrôlées dans la saisie de donnée.

- 2) le nombre d'étudiants qui doivent composer chacune des scéances est calculé en fonction du nombre de scéances n et du nombre d'inscrits nb-ins(i). Pour chaque cours, 2 variables suffisent pour mémoriser le nombre d'étudiants pour chacune des scéances et ce quel que soit le nombre de scéances d'orale :
  - nb-et-sc(i) : contient le nombre minimum d'étudiants qui composeront chaque scéance
  - nbre-sc(i) : contient le nombre de scéances dont le nombre étudiants la composant = nb-et-sc(i) + 1
$$\text{nb-et-sc}(i) = \text{nb-ins}(i) \text{ div } n$$

$$\text{nbre-sc}(i) = \text{nb-ins}(i) \bmod n$$

Exemples1)  $n = 8$ 

$$\text{nb-ins}(i) = 54$$

$$\text{d'où nb-et-sc}(i) = 54 \text{ div } 8 = 6$$

$$\text{nbre-sc}(i) = 54 \text{ mod } 8 = 6$$

cela signifie en pratique que

- les 6 premières scéances orales contiendront 7 étudiants
- les 2 dernières scéances orales contiendront 6 étudiants

2)  $n = 2$ 

$$\text{nb-ins}(i) = 17$$

$$\text{d'où nb-et-sc}(i) = 17 \text{ div } 2 = 8$$

$$\text{nbre-sc}(i) = 17 \text{ mod } 2 = 1$$

cela signifie en pratique que

- la première scéance orale contiendra 9 étudiants
- la deuxième scéance orale contiendra 8 étudiants

Remarques

- Avec cette méthode de calcul, le paramètre d'entrée  $\text{nb-et}(i)$  est considéré comme le nombre maximum d'étudiants qui peuvent composer une scéance.
- Cette méthode assure une répartition parfaitement équilibrée des étudiants inscrits sur toutes les scéances d'orale.

Conclusion

Après avoir appelé la procédure  $\text{const-scéance}$ , on a les informations suivantes

- $\text{nb-scea}$  : nombre de sommets du graphe
- pour chaque cours  $i$  :
  - $\text{c-c}(i,1)$  : le sommet correspondant à l'épreuve écrite du cours  $i$
  - $\text{c-c}(i,2)$  : le sommet correspondant à la première épreuve orale
  - $\text{c-c}(i,3)$  : le nbre de sommets correspondants à des scéances d'épreuve orale du cours  $i$
- $\text{nb-et-sc}(i)$  et  $\text{nbre-sc}(i)$  qui permettent de déterminer exactement le nombre d'étudiants qui composeront chaque sommet.

Ces informations suffisent pour attribuer à chaque sommet leurs différents composants (cours, professeur, étudiant). Cette attribution de composants est vue dans le sous-paragraphe qui suit.



## I.2.2. Détermination des composants d'un sommet

=====

Un sommet  $x$  est un sous-ensemble de  $\xi \cup \mathcal{P} \cup \mathcal{C}$

avec  $\xi = \{e_i \text{ t.q. } i \in I_1\}$  l'ensemble des étudiants  
 $\mathcal{P} = \{p_j \text{ t.q. } j \in I_2\}$  l'ensemble des professeurs  
 $\mathcal{C} = \{c_k \text{ t.q. } k \in I_3\}$  l'ensemble des cours

Le problème consiste donc à attribuer à chaque sommet  $x(s)$  ( $1 \leq s \leq \text{nb-scea}$ ) un sous-ensemble de  $\xi \cup \mathcal{P} \cup \mathcal{C}$

## I.2.2.1. Attribution du cours

=====

Par convention, un sommet  $x(s)$  ne contient qu'un seul cours. En effet, une épreuve d'examen ne porte en général que sur la matière d'un seul cours. Par construction (voir sous paragraphe précédent), à chaque sommet du graphe correspond un et un seul cours. D'où si au sommet  $x(s)$  correspond le cours  $c_{s_i}$  alors on attribue  $c_{s_i}$  à  $x(s)$  :  $c_{s_i} \in x(s)$ .

De plus, pour faire la distinction entre un sommet correspondant à une épreuve écrite et un sommet correspondant à une épreuve orale,

dans le 1er cas on notera :  $c_{s_i}^* \in x(s)$

dans le 2ème cas on notera :  $c_{s_i} \in x(s)$

Cette distinction s'impose car le processus de coloration est différent suivant que le sommet coloré correspond à une épreuve écrite ou orale.

## I.2.2.2. Attribution du professeur

=====

Par convention, un sommet  $x(s)$  ne contient qu'un seul professeur. En effet, l'épreuve d'examen ne porte que sur la matière d'un seul cours et un cours n'est en général sous la responsabilité que d'un seul professeur. D'où si au sommet  $x(s)$  correspond le cours  $c_{s_i}$  et si le cours  $c_{s_i}$  est sous la responsabilité du professeur  $p_{s_{i_s}}$  alors, on attribue  $p_{s_{i_s}}$  à  $x(s)$

$p_{s_{i_s}} \in x(s)$

## I.2.2.3. Attribution des étudiants

=====

Si l'attribution du cours et du professeur à un sommet est évidente et complètement prédéterminée, par contre l'attribution d'étudiants à un sommet n'est pas complètement prédéterminée.

Exemple

Soit un sommet  $s$  correspondant à une épreuve orale du cours n°  $i$ .

si  $\text{nb-ins}(i) = n$

si le nombre d'étudiants devant composer le sommet  $s = p$

alors il existe  $\frac{n!}{p!(n-p)!}$  possibilités différentes d'attribution d'étudiants au sommet  $s$ .

Par contre, il existe des critères à postériori pour détecter si une attribution d'étudiants est bonne ou non. Avant de définir ces critères, nous allons poser le problème en fixant les conditions de sortie après l'attribution. Cette manière de poser le problème n'est pas unique mais permet de fixer facilement les critères de jugement d'une attribution.

## Préconditions

=====

On a en entrée : -  $\text{nb-scea}$  sommets  $x(s)$   $1 \leq s \leq \text{nb-scea}$

- pour chaque sommet  $x(s)$ 
  - un nombre d'étudiants à attribuer  $n(s)$
  - un ensemble d'étudiants qui peuvent être attribués à ce sommet  $x(s)$  :  $e_i$  peut être attribué à  $x(s)$  si  $e_i$  est inscrit au cours  $j$  correspondant à  $x(s)$

## Postconditions

=====

On a en sortie : - pour chaque sommet  $x(s)$

- $n(s)$  étudiants attribués. si  $e_i$  est attribué au sommet  $x(s)$  alors  $e_i \in x(s)$
- une matrice  $t\text{-simul}(1..\text{nb-scea}, 1..\text{nb-scea})$  t.q.
  - pour tout  $1 \leq i, j \leq \text{nb-scea}$ 

$$t\text{-simul}(i, j) = 1 \text{ ssi } \exists k \in I_1 \text{ t.q. } e_k \in x(i) \text{ et } e_k \in x(j)$$

$$= 0 \text{ sinon}$$



$t\text{-simul}(i,j) = 1 \iff$  le sommet  $x(i)$  et le sommet  $x(j)$  ont au moins un étudiant commun

$t\text{-simul}(i,j) = 1 \implies$  le sommet  $x(i)$  et le sommet  $x(j)$  ne pourront se voir attribuer une même couleur lors du processus de coloration.

#### Critères de jugement d'une attribution

=====

- On pourrait prendre comme critère le nombre  $s = \sum_i \sum_j t\text{-simul}(i,j)$
  - autre critère : soit  $s = \max_i \sum_j t\text{-simul}(i,j)$
  - autre critère : soit  $s = \sum_i \sum_j t\text{-simul}(i,j) \times \max_i \sum_j t\text{-simul}(i,j)$
- Dans les trois cas, une attribution A étant considérée meilleure qu'une attribution B si  $s(A) \leq s(B)$ .

Le problème étant bien posé et les critères de choix d'une attribution étant bien définis, il nous reste maintenant à trouver une attribution d'étudiants qui soit la meilleure possible (c'est-à-dire celle qui minimise s)

1ere idée : essayer toutes les possibilités d'attribution et conserver celle qui minimise s.

Avantage : on est assuré que l'attribution choisie est la "meilleure".

Inconvénient : le nombre de possibilité est énorme.

Exemple : 10 cours, 100 étudiants, 10 scéances par cours

pour chaque cours, on a  $\frac{100!}{90! 10!}$  possibilités

d'où globalement on a  $\left( \frac{100!}{90! 10!} \right)^{10}$  possibilités d'attribution à tester

Conclusion : idée à rejeter

2ème idée : n'essayer qu'une possibilité dont on sait par construction qu'elle donnera une solution acceptable.

Avantage : temps d'exécution rapide.

Inconvénient : on n'est pas sûr que l'attribution choisie est la meilleure

Conclusion : idée à approfondir.

## Méthode de base pour l'attribution des étudiants

=====

Principe de "l'alphabet" : je choisis un étudiant  $e_i$  parmi les étudiants non encore attribués

- pour tous les cours  $c$  de n°  $j$  tel que  $e_i$  est inscrit à  $c$  de n°  $j$  faire
  - si  $c-c(j,1) \neq 0$  alors  $j$ 'attribue  $e_i$  au sommet correspondant à l'épreuve écrite du cours  $c$
  - si  $c-c(j,2) \neq 0$  alors  $j$ 'attribue  $e_i$  au sommet correspondant à la première séance orale de ce cours non encore complètement remplie en étudiants.
- pour chaque sommet  $s, s'$  tel que  $e_i \in s$  et  $e_i \in s'$  :  $t\text{-simul}(s, s') = 1$

## A. Justification de cette méthode

=====

Cette méthode se justifie par la propriété suivante :

soit  $s(k), \dots, s(k+1)$  sommets correspondant à toutes les séances orales du cours  $c_i$

soit  $s(k'), \dots, s(k'+1')$  sommets correspondant à toutes les séances orales du cours  $c_i$ ,

soit  $0 \leq j \leq 1$      $0 \leq j' \leq 1'$

1) si  $t\text{-simul}(k+j, k'+j') = 1$   
 alors pour tout  $0 \leq n \leq j$      $j' < n' \leq 1'$   
 $t\text{-simul}(k+n, k'+n') = 0$  et  $t\text{-simul}(k+n, k'+n) = 0$

Démonstration : par l'absurde

=====

soit  $0 \leq n \leq j$  et  $j' < n' \leq 1'$  t.q.  $t\text{-simul}(k+n, k'+n') = 1$

d'où  $\exists e' \in \mathcal{E}$  t.q.  $e' \in s(k+n)$  et  $e' \in s(k'+n')$ .

Or,  $t\text{-simul}(k+j, k'+j') = 1$

d'où  $\exists e \in \mathcal{E}$  t.q.  $e \in s(k+j)$  et  $e \in s(k'+j')$ .

$e \in s(k+j)$   $e' \in s(k+n)$   $n < j \Rightarrow e'$  a été choisi avant  $e$  (1)

$e \in s(k'+j')$   $e' \in s(k'+n')$   $n' > j' \Rightarrow e'$  a été choisi avant  $e$  (2)

(1) et (2) incompatibles d'où  $t\text{-simul}(k+n, k'+n') = 0$

$t\text{-simul}(k'+n', k+n) = 0$  par symétrie.



- 2) si  $t\text{-simul}(k + j, k' + j') = 1$   
 alors pour tout  $j < n \leq 1 \quad 0 \leq n' < j'$   
 $t\text{-simul}(k + n, k' + n') = 0$  et  $t\text{-simul}(k' + n', k + n) = 0$

Démonstration : par l'absurde

=====

soit  $0 \leq n' < j'$  et  $j < n \leq 1$  t.q.  $t\text{-simul}(k + n, k' + n') = 0$

d'où  $\exists e' \in E$  t.q.  $e' \in s(k + n)$  et  $e' \in s(k' + n')$ .

Or,  $t\text{-simul}(k + j, k' + j') = 1$

d'où  $\exists e \in E$  t.q.  $e \in s(k + j)$  et  $e \in s(k' + j')$ .

$e \in s(k + j)$   $e' \in s(k + n)$   $n > j \Rightarrow e'$  a été choisi après  $e$  (1)

$e \in s(k' + j')$   $e' \in s(k' + n')$   $n' < j' \Rightarrow e'$  a été choisi avant  $e$  (2)

(1) et (2) incompatibles d'où  $t\text{-simul}(k + n, k' + n') = 0$

Conséquence de cette méthode

=====

Isolons de la matrice  $t\text{-simul}$  toutes les lignes correspondant aux oraux du cours  $c_i$  et toutes les colonnes correspondant aux oraux du cours  $c_j$

( $i \neq j$ ), si  $c - c(i, 3) = n$ ,  $c - c(j, 3) = p$ , on obtient une sous-matrice

$A$   $n \times p$  alors si  $A(s, t) = 1 \quad 1 \leq s \leq n \quad 1 \leq t \leq p$

on a une sous-matrice de ce type :

|     |       |     |       |
|-----|-------|-----|-------|
|     |       | $t$ |       |
|     |       |     | $O_1$ |
| $s$ |       | 1   |       |
|     | $O_2$ |     |       |

propriété n° 1  $\Rightarrow O_1$

propriété n° 2  $\Rightarrow O_2$

Conséquence :

$$\sum_s \sum_t A(s, t) < n + p$$

Démonstration :

Considérons d'un façon imagée la matrice  $A$  comme un chemin :

$A(i, j) = 1$  si le chemin passe par  $(i, j)$

soit  $(i_1, j_1) \quad 1 \leq i_1 \leq n \quad 1 \leq j_1 \leq p$  t.q.

1)  $\forall 0 < i < i_1, \quad 1 \leq j \leq p \quad A(i, j) = 0$

2)  $A(i_1, j_1) = 1$

3)  $1 \leq j < j_1, \quad 1 \leq i \leq n \quad A(i, j) = 0.$





Considérons l'algorithme suivant qui parcourt ce chemin

### Algorithme

=====

- 1)  $s = 1, (i, j) = (i_1, j_1)$
- 2) si  $(i, j) = (i_n, j_p)$  alors arrêter  
sinon 1')  $s = s + 1$   
2') choisir  $(i_c, j_c)$  tel que
  - 1)  $i_c \geq i \quad j_c \geq j \quad (i_c, j_c) \neq (i, j)$
  - 2)  $A(i_c, j_c) = 1$
  - 3)  $\forall i' \leq i' \leq i_c \quad \forall j' \leq j' \leq j_c$   
 $(i', j') \neq (i, j) \text{ ou } (i', j_c) \neq (i_c, j_c) \Rightarrow A(i', j') = 0$
- 3')  $(i, j) = (i_c, j_c)$
- 4') retour en 2)

### Remarque

=====

|       |     |       |
|-------|-----|-------|
|       | $j$ | $j_c$ |
| $j$   | 1   | 0     |
|       | 0   | 0     |
| $i_c$ | 0   | 1     |

### Lemme

=====

$(i, j)$  tel que  $A(i, j) = 1$ , alors l'algorithme passe une et une seule fois par  $(i, j)$

### Démonstration

=====

- 1) par définition de  $(i_1, j_1)$ , l'algorithme assure qu'il "n'oublie" aucun  $(i, j)$  tel que  $i \leq i_1$  ou  $j \leq j_1$ ,  $(i, j) \neq (i_1, j_1)$  et  $A(i, j) = 1$
- 2) par définition de  $(i_n, j_p)$ , l'algorithme assure qu'il "n'oublie" aucun  $(i, j)$  tel que  $i > i_n$  ou  $j > j_p$  et  $A(i, j) = 1$
- 3) Soit  $(i', j')$  tel que  $A(i', j') = 1$  et  $i_1 < i' < i_n \quad j_1 < j' < j_p$   
Supposons que l'algorithme a oublié  $(i', j')$

### Conséquence

=====

Il a fallu que la situation suivante en 2) de l'algorithme ait été présente

|       | $j_1$ |  | $j$ |   | $j_c$ |   |
|-------|-------|--|-----|---|-------|---|
| $i$   |       |  |     |   |       |   |
|       |       |  |     | 1 | 0     | 0 |
| $i'$  |       |  |     | 0 | 0     | 0 |
|       | 1     |  |     | 0 | 0     | 0 |
|       |       |  |     | 0 | 0     | 0 |
| $i_c$ |       |  |     | 0 | 0     | 1 |
|       |       |  |     |   |       |   |

I.17

Cette situation est impossible vu la propriété 2) d'où la supposition est absurde.

- 4) Vu que  $(i_c, j_c)$  est choisi tel que  $i_c \geq i$   $j_c \geq j$   $(i_c, j_c) \neq (i, j)$   
l'algorithme ne passe qu'une seule fois au maximum par tout  $(i, j)$

Conséquence

=====

Après exécution de l'algorithme  $s = \sum_{i=1}^n \sum_{j=1}^p A(i, j)$

Or,  $s$  = nombre de fois que l'on passe par le 2) de l'algorithme,

l'algorithme assurant une progression "à droite" et "vers le bas"

dans le chemin joignant  $(i, j_1)$  à  $(i_n, j_p)$ , il s'en suit que le nombre de fois que l'on passe par le 2) de l'algorithme =  $s \leq (i_n - i_1)$

+  $(j_p - j_1) + 1$

$$\left. \begin{array}{l} i_n - i_1 \leq n - 1 \\ j_p - j_1 \leq p - 1 \end{array} \right\} \text{ d'où } s \leq n - 1 + p - 1 + 1$$

$$\text{d'où } s = \sum_{i=1}^n \sum_{j=1}^p A(i, j) < n + p \quad \text{cqfd}$$

Exemple :  $n = p = 8$

alors sur 64 "1" possible, il n'y en a au maximum, dans le plus mauvais des cas, que 15. Or, il est bien évident que l'ordre dans lequel les étudiants sont choisis peut encore diminuer le nombre  $s$ .



### B. Critère de choix des étudiants

=====

Principes de base : 1) si  $t\text{-simul}(i,j) = 1$ , il faut maximaliser le  
 ===== nombre d'étudiants  $e$  tel que  $e \in s(i)$  et  $s(j)$

Résultat idéal : Soit  $s(i)$  et  $s(j)$   
 =====  $n(i)$  et  $n(j)$   $n(i) \leq n(j)$

alors si  $t\text{-simul}(i,j) = 1 \Rightarrow$  les  $n(i)$  étudiants attribués à  
 $s(i)$  le sont également à  $s(j)$

2) si un étudiant  $e$  doit être attribué à  $n$  scéances,  
 alors, cet étudiant  $e$  va mettre  $n^2$  éléments de  
 $t\text{-simul}$  à 1. Il faut parmi ces  $n^2$  éléments  
 maximaliser le nombre d'éléments qui étaient  
 déjà à 1.

### Résultat idéal.

=====

Soit  $s(i)$  et  $s(j)$  tel que  $e \in s(j)$  et  $e \in s(i)$   
 alors  $t\text{-simul}(i,j) = 1$  même en retirant  $e$  de  $s(i)$  et  $s(j)$

### Méthode de base

=====

On considère, à un moment donné, tous les sommets correspondant à des  
 scéances en cours de remplissage (c-à-d., au moins 1 étudiant déjà attri-  
 bué et au moins 1 étudiant à encore attribuer). Le choix va se porter  
 sur l'étudiant qui, s'il est choisi, minimisera le nombre d'ajout de 1  
 sur les lignes correspondant à ces sommets tout en maximalisant le nombre  
 de sommets auxquels il sera attribué.

### Algorithmes

=====

Définissons d'abord les variables suivantes :

- 1) num = nombre d'étudiants déjà attribués
- 2) nb-étud = nombre d'étudiants à choisir
- 3) k = nombre de sommets correspondant à des oraux en cours de remplissage
- 4)  $\text{étud}(i,1)$  = nombre de sommets auxquels l'étudiant  $e_i$  peut être attri-  
 bué et qui sont en cours de remplissage
- 5)  $\text{étud}(i,2)$  = nombre de sommets auxquels l'étudiant  $e_i$  peut être attri-  
 bué et qui ne sont pas en cours de remplissage

6)  $\text{étud}(i,3)$  = nombre de sommets qui, en cas d'attribution de  $e_i$ , seraient totalement remplis

7)  $\text{étud}(i,4)$  = nombre minimum d'étudiants manquant à chaque sommets auxquels  $e_i$  serait attribué.

Les algorithmes des pages suivantes n'assurent pas que l'ordre est optimal. Toutefois, il représentent une façon "naturelle" de choisir l'étudiant en jouant sur les critères matérialisés par  $\text{étud}(i,j)$   $1 \leq j \leq 4$  et devraient normalement donner des meilleurs résultats en moyenne qu'un ordre de choix arbitraire.

Algorithme 1  
=====

- 1) Parmi les étudiants  $e_i$  ( $\text{num} < i \leq \text{nb-étud}$ )  
 Si  $k \leq n\text{-min}$  prendre celui tel que
  - 1)  $\text{étud}(i,1)$  est maximum
  - 2)  $\text{étud}(i,2)$  est maximum
 Sinon prendre celui tel que
  - 1)  $\text{étud}(i,2)$  est minimum
  - 2) si  $\text{étud}(i,2) = 0$ 
    - celui t.q. 1)  $\text{étud}(i,3)$  est minimum
    - 2)  $\text{étud}(i,1)$  est maximum
  - sinon celui t.q. 1)  $\text{étud}(i,1)$  est maximum
  - 2)  $\text{étud}(i,3)$  est maximum
- 2) mettre à jour  $k$ ,  $t\text{-simul}$ , les sommets auxquels  $e_i$  est attribué
- 3) intervertir  $e_i$  et  $e_{\text{num}}$

Algorithme 1'  
=====

- 1) Parmi les étudiants  $e_i$  ( $\text{num} < i \leq \text{nb-étud}$ )  
 prendre celui tel que
  - 1)  $\text{étud}(i,2)$  est minimum
  - 2) si  $\text{étud}(i,2) = 0$ 
    - celui tel que 1)  $\text{étud}(i,3)$  est minimum
    - 2)  $\text{étud}(i,1)$  est maximum
  - sinon celui tel que 1)  $\text{étud}(i,1)$  est maximum
  - 2)  $\text{étud}(i,3)$  est maximum
- 2) mettre à jour  $k$ ,  $t\text{-simul}$ , les sommets auxquels  $e_i$  est attribué
- 3) intervertir  $e_i$  et  $e_{\text{num}}$



## Algorithme 2

=====

- 1) Parmi les étudiants  $e_i$  ( $\text{num} < i \leq \text{nb-étud}$ )
  - si  $k \leq n\text{-min}$  prendre celui tel que
    - 1)  $\text{étud}(i,1)$  est maximum
    - 2)  $\text{étud}(i,2)$  est maximum
  - sinon prendre celui tel que
    - 1)  $\text{étud}(i,2)$  est minimum
    - 2) si  $\text{étud}(i,2) = 0$ 
      - celui tel que 1)  $\text{étud}(i,3)$  est minimum
      - 2)  $\text{étud}(i,1)$  est maximum
      - 3)  $\text{étud}(i,4)$  est maximum
    - sinon celui tel que 1)  $\text{étud}(i,1)$  est maximum
    - 2)  $\text{étud}(i,3)$  est maximum
- 2) mettre à jour  $k$ ,  $t\text{-simul}$ , les sommets auxquels  $e_i$  est attribué
- 3) intervertir  $e_i$  et  $e_{\text{num}}$

## Algorithme 3

=====

- 1) parmi les étudiants  $e_i$  ( $\text{num} < i \leq \text{nb-étud}$ )
  - si  $k \leq n\text{-min}$  prendre celui tel que
    - 1)  $\text{étud}(i,2) + \text{étud}(i,1)$  est maximum
    - 2)  $\text{étud}(i,1)$  est maximum
  - sinon prendre celui tel que
    - 1)  $\text{étud}(i,2)$  est minimum
    - 2) si  $\text{étud}(i,2) = 0$ 
      - celui tel que 1)  $\text{étud}(i,1)$  est maximum
      - 2)  $\text{étud}(i,3)$  est minimum
      - 3) si  $\text{étud}(i,3) = 0$ 
        - celui t.q.  $\text{étud}(i,4)$  est maximum
        - sinon celui t.q.  $\text{étud}(i,4)$  est minimum
      - sinon celui tel que 1)  $\text{étud}(i,3)$  est maximum
      - 2)  $\text{étud}(i,1)$  est maximum
  - 2) mettre à jour  $k$ ,  $t\text{-simul}$ , les sommets auxquels  $e_i$  est attribué
  - 3) intervertir  $e_i$  et  $e_{\text{num}}$

## Conclusion

=====

Après avoir appelé la procédure implémentant un des algorithmes d'attribution d'étudiants, on dispose des informations suivantes :

- nb-scea : nombre de sommets du graphes
- pour chaque sommet  $s(i)$   $1 \leq i \leq \text{nb-scea}$ 
  - le cours correspondant à ce sommet
  - le professeur correspondant à ce sommet
  - les étudiants correspondant à ce sommet
- t-simul, matrice à 2 dimensions tel que pour tout  $1 \leq i, j \leq \text{nb-scea}$  si  $t\text{-simul}(i, j) = 1 \iff s(i)$  et  $s(j)$  comprennent au moins un étudiant en commun

Ces informations sont suffisantes pour que l'on puisse entamer le processus de coloration de ce graphe.

## I.3. Première approche du processus de coloration du graphe

=====

Nous disposons à l'heure actuelle d'un ensemble  $E = \{U, P, C\}$

$X \subseteq \{x \text{ t.q. } x \text{ sous-ensemble de } E\}$  ou  $X$  est l'ensemble des sommets

$U = \{(x, y) \text{ t.q. } x, y \in X, x \neq y, x \cap y \neq \emptyset\}$

$H = \{h \text{ t.q. } h \in H\}$  l'ensemble des couleurs

En effet : -  $U, P$  et  $C$  peuvent être considérés comme paramètre d'entrée

- $X$  l'ensemble a pu être déterminé à partir de  $E$  (voir paragraphe précédent)

-  $(x, y) \in U \iff \{t\text{-simul}(x, y) = 1 \text{ ou } \exists p \in P \text{ t.q. } p \in x \text{ et } p \in y \text{ ou } \exists c \in C \text{ t.q. } c \in x \text{ et } c \in y\}$

d'où l'ensemble  $U$  est également complètement déterminé.

- $H$  peut être considéré comme paramètre d'entrée.



Dans une première approche, le processus de coloration consistera en l'implémentation d'une fonction  $f$ .

$$f : X \rightarrow H : x \rightarrow f(x) = h$$

fonction 1) complètement définie sur  $X$

$$2) (x,y) \in U \Rightarrow f(x) \neq f(y)$$

La méthode de coloration qui serait utilisée en se limitant uniquement à la contrainte 2) ci-dessus est la méthode saturation dont je rappelle ci-dessous l'algorithme.

Algorithme sat

=====

- 1) Ordonner tous les sommets selon la notion de degré maximal  
c-à-d.  $d_1 \geq d_2 \dots \geq d_n$
- 2) colorer le sommet  $x_1$  avec la première couleur  
interdire cette couleur pour les adjacents de  $x_1$
- 3) prendre le sommet dont le degré de saturation est le plus élevé parmi les sommets non colorés (en cas d'égalité utiliser l'ordre de 1)
- 4) colorer ce sommet avec la première couleur possible et interdire cette couleur à ses adjacents.
- 5) s'il reste des sommets à colorer, aller en 3)  
sinon arrêter

N.B. :  $d_i$  = le nombre de sommets adjacents aux sommets  $x_i$   
 degré de saturation  $(x) = \# \{ h \text{ t.q. } \exists y \in X, (x,y) \in U, f(y) = h \}$

C H A P I T R E      I I



## Chapitre II : L'horaire d'examen complet comme modèle de graphe avec contrainte de temps

=====

Nous adaptons le modèle de graphe aux contraintes pédagogiques et nous décrivons les fichiers relatifs aux contraintes directes et indirectes.

### II.1. Le modèle de graphe et les contraintes directes

=====

Comme on vient de le voir au premier chapitre, le calcul d'horaire d'examen revient à donner une demi-journée à chaque scéance d'examen tout en respectant les contraintes dites indirectes (aucun étudiant, aucun professeur n'est pris deux fois à la même demi-journée). Le modèle de graphe correspondant (I 1.2) résout parfaitement ce problème et le calcul d'horaire se ramène à la coloration d'un graphe simple.

Ce modèle ne satisfait pourtant pas toutes les exigences réelles : il faut lui ajouter les contraintes pédagogiques dites contraintes directes.

#### II.1.1. Les contraintes directes

=====

Elles représentent les exigences qui se posent souvent dans le calcul d'horaire d'un établissement scolaire et qui doivent donc être envisagées au niveau du modèle de graphe. Nous distinguons 10 type de contraintes :

- 1) Dimanche : Pour toutes les scéances d'examen, empêcher qu'une scéance puisse se dérouler un dimanche (pour des raisons évidentes).
- 2) Samedi : Pour un cours donné, empêcher qu'une scéance d'examen portant sur ce cours puisse se dérouler un samedi (ex. : un professeur ne désirant pas interroger un samedi).
- 3) Indisponibilités de professeurs : un certain professeur étant absent certains jours pour diverses raisons (voyage, travail ...)

- 4) Succession des épreuves : Pour un même étudiant, il faut assurer un certain délai de "reflexion" entre deux examens consécutifs.
- 5) Consécutivité des oraux : les professeurs préfèrent que les séances d'orale d'un même examen se déroulent dans un laps de temps minimum.
- 6) Succession écrit-oral : un examen écrit portant sur un cours doit se dérouler avant l'oral portant sur ce cours.
- 7) Délai entre écrit et oral : Un délai peut être exigé entre l'épreuve orale et l'épreuve écrite. Il peut être maximum (par exemple : écrit le matin, oral en déans les deux jours qui suivent) ou minimum ( ex. : en vue de la correction de l'écrit).
- 8) Jour d'empêchement pour un cours : pour diverses raisons, on peut interdire certaines périodes de la session à un cours.
- 9) Examen fixé à l'avance : la date d'une épreuve peut être fixée à l'avance (ex. : pour un professeur venant de l'extérieur).
- 10) Date de début et de fin de session : bien entendu, l'établissement de l'horaire se fait dans un cadre temporel qui est la session.

Toutes ces contraintes doivent être envisagées dans le modèle de graphe qu'on doit par conséquent améliorer en lui ajoutant le concept de fonction de temps (ou matrice de temps). Ces modifications font l'objet du paragraphe suivant.



## II.1.2. Graphe avec fonction de temps

=====

Reprenons le modèle I.1.2.

soient  $E = \{e_1, e_2, e_3, e_4, p_1, p_2, c_1, c_2\}$ 

$$X \subseteq \{x \text{ t.q. } x \text{ sous-ensemble de } E\} \text{ où } X \text{ est l'ensemble des sommets du graphe}$$

$$U = \{(x,y) \text{ t.q. } x,y \in X, x \neq y, x \cap y \neq \emptyset\} \text{ où } U \text{ est l'ensemble des arcs du graphe}$$

$$H = \text{ensemble des couleurs possibles (demi-journées de la session)}$$
Ils définissent le graphe  $G(X,U)$ Nous ajoutons à ce graphe une fonction de temps

$$L : X \times H \longrightarrow \mathbb{N} \text{ telle que}$$

$$L(x,h) \begin{cases} = 0 & \text{si le sommet } x \text{ peut prendre la couleur } h \\ \neq 0 & \text{sinon} \end{cases}$$

Cette fonction de temps contiendra au début de la coloration une partie des contraintes directes et sera modifiée au fur et à mesure que l'algorithme de coloration avance.

La fonction de temps peut être mise sous forme d'une matrice de temps à  $(nb-scea \times durée)$  éléments

où  $nb-scea$  = nombre de sommets

durée = nombre de couleurs possibles (1/2 journées)

Nous allons voir comment la matrice de temps est initialisée avec une partie des contraintes directes

Soit  $E = \{e_1, e_2, e_3, e_4, p_1, p_2, c_1, c_2\}$

$$X = \left\{ \begin{aligned} x_1 &= \{c_1^*, p_1, e_1, e_2, e_3, e_4\} \\ x_2 &= \{c_1, p_1, e_1, e_2\} \\ x_3 &= \{c_1, p_1, e_3, e_4\} \\ x_4 &= \{c_2^*, p_1, e_1, e_2\} \\ x_5 &= \{c_2, p_1, e_1\} \end{aligned} \right\}$$

Soit durée = 20 et couleur 1 correspond à un lundi matin

- début de la session = 5  
 Pour s allant de 1 à 5  
 Pour j allant de 1 à 4 :  $L(s,j) = 999$
- fin de la session = 17  
 Pour s allant de 1 à 5  
 Pour j allant de 18 à durée :  $L(s,j) = 999$
- dimanche (contrainte 1)  
 Pour s allant de 1 à 5  
 $L(s,13) = 999$   
 $L(s,14) = 999$
- le cours  $c_1$  ne peut avoir examen un samedi  
 Pour s allant de 1 à 3  
 $L(s,11) = 999$   
 $L(s,12) = 999$
- Le professeur  $p_2$  est absent la demi-journée 8  
 Pour s allant de 4 à 5  
 $L(s,8) = 999$
- Délai minimum = 2 pour le cours  $c_1$  entre l'écrit et l'oral  
 Pour s allant de 2 à 3  
 Pour j allant de 5 à 7 :  $L(s,j) = 999$   
 Pour j allant de 14 à 17 :  $L(1,j) = 999$   
 ( $x_1$  pouvant se dérouler au plus tôt en 5,  $x_2$  et  $x_3$  ne peuvent donc se dérouler en 5,6,7)  
 ( $x_2$  ou  $x_3$  ne pouvant se dérouler qu'au plus tard en 16,  $x_1$  ne peut se dérouler en 14, 15, 16 et 17).

On constate que la matrice de temps est initialisée en tenant compte au maximum des contraintes directes et indirectes. Lorsque cette initialisation est terminée, le processus de coloration de graphe peut commencer avec le maximum de sécurité possible.



## II.2. Algorithme de coloration exploitant la matrice de temps

=====

- Un algorithme de coloration de graphe consiste en la matérialisation d'une fonction  $f : X \longrightarrow H : x \longrightarrow h$   
avec  $f$  partout définie sur  $X$   
Si l'on ajoute dans le graphe  $X$  la notion d'adjacence entre sommets, on introduit un ensemble  $U = \{ (x,y) \text{ t.q. } x \in X, y \in X \}$
- En se limitant aux contraintes indirectes, on ajoute une propriété à  $f$  si  $(x,y) \in U$  alors  $f(x) \neq f(y)$   
 $x \neq y$
- En introduisant une matrice de temps  $L$ , matrice à 2 dimension tel que
  - une ligne de  $L$  correspond à un sommet de  $X$
  - une colonne de  $L$  correspond à une couleur de  $H$
 On peut définir une correspondance entre la fonction  $f$  et la matrice  $L$   
Cette correspondance sera assurée par les propriétés suivantes :
  - 1)  $f(x_i) = h_j \implies L(i,j) = 0$
  - 2)  $(x_i, y_k) \in U \quad f(x_i) = h_j \implies L(k,j) \neq 0$
- En ajoutant la notion de contraintes directes, on complique encore les éliminations dans la matrice  $L$ , éliminations qui portent non seulement sur la colonne correspondant à la couleur choisie, mais aussi sur d'autres colonnes.

Un algorithme de coloration peut s'écrire de façon générale :

- 1) prendre un sommet  $x$  non colorié
- 2) donner une couleur  $h$  à ce sommet
- 3) faire les éliminations dues aux contraintes
- 4) s'il reste des sommets à colorier aller en 1)  
sinon arrêter.

A la lecture de cet algorithme, trois problèmes se posent :

- 1) quel sommet choisir en 1) ?
- 2) quelle couleur donner à ce sommet ?
- 3) que faire si ce sommet ne peut plus prendre de couleur ?

Ces 3 problèmes font l'objet des 3 sous-paragraphe qui suivent et qui traceront les grandes lignes de l'algorithme final.

## II. 2.1. Choix du sommet à colorier

=====

Soient au départ nb-scea sommets à colorier

- nb-placé sommets déjà coloriés
- n (=nb-scea - nb-place) sommets  $x_1, \dots, x_n$  non encore coloriés.
- une matrice L tel que  $1 \leq i \leq n$   $1 \leq h \leq \#H$   
 si  $L(i, h) = 0$  alors le sommet  $x_i$  peut encore prendre la couleur h

Pour chaque sommet  $x_i$   $1 \leq i \leq n$  définissons

- npos( $x_i$ ) = nombre de couleurs que peut encore prendre le sommet  $x_i$
- d( $x_i$ ) = nombre de sommets adjacents à  $x_i$  et non encore placés.

C'est bien évidemment sur ces deux valeurs que les critères de choix d'un sommet vont jouer.

Ces critères peuvent être :

- choisir  $x_i$  tel que npos ( $x_i$ ) est minimum
- choisir  $x_i$  tel que d( $x_i$ ) est maximum
- choisir  $x_i$  tel que npos ( $x_i$ ) - d( $x_i$ ) est minimum

Dans le cas du calcul d'un horaire, pour chaque sommet  $x_i$  correspondant à une scéance d'examen s, on calcule deux variables.

- npos( $x_i$ ) = nombre de demi-journées qui peuvent encore être attribuées à la scéance s
- norale ( $x_i$ ) = 0 si la scéance s est une scéance d'écrit  
 = le nombre de scéances orales correspondant au même cours sinon

Critère de choix du sommet

=====

- 1) S'il existe des sommets  $x_i$   $1 \leq i \leq n$  tel que npos( $x_i$ )  $\leq$  n-min  
 prendre celui tel que 1) npos( $x_i$ ) est minimum  
 2) norale( $x_i$ ) est maximum
- sinon prendre celui tel que  
 1) norale( $x_i$ ) est maximum  
 2) npos ( $x_i$ ) est minimum



## Justification

=====

Priorité est d'abord donné aux sommets dont le nombre de demi-journées encore attribuable est inférieur à un nombre limite.

Si le nombre de demi-journées encore attribuable est supérieure à ce nombre limite, priorité est faite alors aux scéances d'orales faisant partie d'un cours qui compte de nombreuses scéances d'orales.

La priorité n°1 assure le détectage des sommets  $x_i$  dont  $n_{pos}(x_i)$  risque à court terme (et si ce n'est déjà fait) de passer à 0.

La priorité n° 2 privilégie les blocs de scéances orales correspondant à un seul et même cours et dont on doit assurer une certaine consécuité, la pratique montrant que cette contrainte est la plus dure à assurer.

## II.2.2. Choix de la couleur

=====

Soit un sommet  $x_i$  tel que  $n_{pos}(x_i) \neq 0$ . Il faut donc attribuer au sommet  $x_i$  une des  $n_{pos}(x_i)$  couleurs possibles.

Une méthode simple serait d'attribuer la première couleur disponible.

Une autre méthode serait de donner un poids  $p_h$  à chaque couleur encore possible pour ce sommet et d'attribuer à ce sommet la couleur dont le poids est minimum. Cette méthode est celle qui est effectivement utilisée, le seul problème étant celui de la détermination d'un poids pour chaque couleur.

## Détermination du poids d'une couleur pour un sommet

=====

Soit  $x_i$  le sommet et  $c_h$  la couleur testée

alors  $p_h$  = le nombre d'éliminations dans la matrice L que provoquerait l'attribution de la couleur  $c_h$  au sommet  $x_i$ .

## Justification

=====

- 1) Cette méthode permet de diriger un sommet en priorité sur certaines couleurs par le fait de manipuler le poids d'une couleur.
- 2) D'un point de vue "horaire d'examen", cette méthode assure une dispersion des séances sur la durée de la session.
- 3) Par rapport à la méthode qui consiste à attribuer la première couleur possible, cette méthode a l'avantage d'être plus sûre car elle attribue au sommet la couleur qui a priori est la meilleure.

## II.2.3. Retour en arrière

=====

Soient nb-placé : nombre de sommets déjà placés

$x_i$  : sommet non encore placé tel que  $npos(x_i) = 0$

Deux cas peuvent se présenter

- 1) nb-placé = 0

dans ce cas, le calcul de l'horaire d'examen tout en respectant les diverses contraintes directes et indirectes débouche sur un échec.

Que faire ? rendre les contraintes moins fortes et recommencer le processus de coloration

- 2) nb-placé  $\neq$  0

dans ce cas, on est certain que  $npos(x_i)$  a été mis à 0 par un des sommets déjà placés (si tel n'était pas le cas, cela signifierait que  $npos(x_i)$  avait initialement la valeur 0, ce qui nous ramènerait au premier cas.

Premier sous-problème : déterminer le dernier sommet placé qui a provoqué une élimination dans la  $i^{\text{ème}}$  ligne de L.

idée : il faut qu'un simple parcours de la  $i^{\text{ème}}$  ligne de L puisse déterminer ce sommet.



Principe : la valeur contenue dans  $L(s,h)$  doit permettre de retrouver le sommet qui, le cas échéant, a provoqué l'élimination de  $L(s,h)$

Méthode utilisée : lorsqu'on attribue une couleur à un sommet  $x_k$  on assigne la valeur  $k$  à  $t\text{-or}(\text{nb-placé})$  et la valeur  $\text{nb-placé}$  à chaque élément  $L(s,h)$  éliminée par cette attribution.

Conséquence : Pour déterminer ce sommet

- 1) prendre le plus grand élément  $k$  de la  $i^{\text{ème}}$  ligne de  $L$  qui soit  $\text{nb-placé}$
- 2) indice du sommet =  $t\text{-or}(k)$

Deuxième sous-problème : décolorer tous les sommets placés après ce sommet. Consiste à requalifier chaque élément  $L(s,h) = j$  en mettant  $L(s,h)$  à 0 ( $k + 1 \leq j \leq \text{nb-placé}$ ) + mise à jour de  $\text{npos}(x_s)$  et  $f(x_s)$

Troisième sous-problème : 1) pour chaque sommet  $x_s (s \neq t\text{-or}(k))$ , requalifier  $L(s,h) = k$  en mettant  $L(s,h)$  à 0 + mise à jour des  $\text{npos}(x_s)$  et  $f(x_s)$

$$2) \text{nb-placé} = k - 1$$

Conclusion : le retour en arrière consiste à revenir à un sommet précédemment coloré et à le colorer d'une autre manière si possible. Ce sommet est le dernier sommet placé adjacent au sommet  $x_i$  tel que  $\text{npos}(x_i) = 0$ . tous les sommets colorés par la suite sont décolorés; ce qui provoque bien sûr diverses mise à jours de variables. Ce retour en arrière assure que toutes les possibilités de coloration auront été testées si le processus débouche sur un échec.

### II.3. Les contraintes : structure des fichiers et implémentation

=====

Nous avons décrit le problème d'horaire d'examens (I.1. et II.1.) en le traduisant sous forme d'un graphe avec fonction de temps. Nous allons analyser dans la suite de façon plus précise les données nécessaires à une formulation et implémentation exacte des algorithmes vus jusqu'à présent en décrivant les fichiers correspondants.

Les informations à traiter se repartissent de la façon suivante :

- a) les classes pour lesquelles l'horaire doit être calculé, la date de début de session, la date de fin de session, les cours pour lesquels des examens intéressent des étudiants de cette classe.
- b) les étudiants pour lesquels on désire savoir à quelles classes ils appartiennent et, parmi les cours de cette classe, à quels cours ils sont inscrits pour passer les examens.
- c) les professeurs pour lesquels on désire obtenir les cours dont ils sont responsables et les dates d'indisponibilités.
- d) les cours pour lesquels divers renseignements sont demandés au sujet de la façon dont les examens se déroulent
  - 1) durée de l'écrit en demi-journées (0,1 ou 2)
  - 2) épreuve orale ?
  - 3) date de l'écrit éventuel fixé à l'avance
  - 4) délai entre l'écrit et l'oral au cas où l'examen consiste en une épreuve écrite et une épreuve orale
  - 5) les dates d'empêchement pour le passage des séances d'examens concernant ce cours
  - 6) samedi.

La manière de représenter ces informations sur fichier et la façon dont elles interviennent dans le processus de coloration seront examinées dans ce paragraphe.



## II.3.1. Les classes

=====

Les classes sont déterminées et identifiées suivant trois valeurs :

- 1) le nom de la faculté à laquelle appartient cette classe  
ex : informatique; chimie; etc..
- 2) le cycle à laquelle appartient cette classe  
ex : candidature, licence, etc..
- 3) l'année : 1<sup>e</sup>, 2<sup>e</sup>, 3<sup>e</sup>

Exemple : troisième licence informatique.

D'où, pour la détermination et l'identification d'une classe, deux petits fichiers auxiliaires (fichiers de référence) sont nécessaires.

a) Fichier faculté dont chaque article se décompose de la façon suivante :

- 1) n° de faculté  $i$  ( $1 \leq i \leq nf$ )
- 2) nom de la faculté

Ce fichier est organisé de façon relative. Le  $n^{\text{ième}}$  article de ce fichier est l'article dont le n° de faculté est égal à la valeur  $n$ .

Au départ, ce fichier est vide ( $\Rightarrow nf = 0$ ). Lorsque un nouvel article est ajouté,  $nf$  est incrémenté de 1. Pour cet article, le n° de faculté =  $nf$  et le nom de la faculté = le nom donné par l'opérateur.

b) Fichier cycle dont chaque article se décompose de la façon suivante :

- 1) n° de cycle  $i$  ( $1 \leq i \leq nc$ )
- 2) nom du cycle

Ce fichier est organisé de façon séquentielle et est initialisé avant l'exécution du programme. Le n° de cycle est identifiant,  $nc$  = nombre de cycles auxquels peut appartenir une classe.

Quand l'opérateur veut identifier la classe sur laquelle il désire travailler, il donne donc successivement

- a) un numéro de faculté     $nof$
- b) un numéro de cycle       $noc$
- c) une année                 $noa$

En concaténant  $nof$ ,  $noc$  et  $noa$ , on obtient un identifiant pour une classe =  
idclasse.

Pour déterminer les dates de début et de fin de session, on le fait par rapport à des dates de références qui sont mises dans un fichier date.

c) Fichier date dont chaque article se décompose de la façon suivante :

- 1) nom du mois
- 2) date de début
- 3) date de fin

Ce fichier est organisé de façon séquentielle et est créé et initialisé avant la première exécution du programme. Deux contraintes existent sur ce fichier (en plus des contraintes de vraisemblance)

- a) la première demi-journée du premier article est un lundi matin
- b) le nombre de demi-journée totale est égal à 114

Exemple : mai      0931

juin      0130

juillet 0104

le 9 mai est un lundi

le nombre de demi-journées entre le 9 mai et le 4 juillet = 114

On peut maintenant décrire complètement un article du fichier classe.

#### Fichier classe

- 1) numéro de la classe  $i$        $1 \leq i \leq \text{nb-classe}$
- 2) idclasse
- 3) date de début de session
- 4) date de fin de session
- 5) les cours de cette classe = 38 numéros

Ce fichier est organisé de façon relative. Le  $i^{\text{ème}}$  article de ce fichier est l'article dont le numéro de la classe =  $i$ .

Une classe est donc identifiée soit par son numéro de classe  
soit par idclasse.



Au départ, ce fichier est vide ( $\Rightarrow$  nb-classe = 0)

Idclasse permet de retrouver la faculté, le cycle et l'année de la classe. La date de début de session est calculée par rapport à la date de la première demi-journée du fichier date.

La date de fin de session est également calculée par rapport à la date de la première demi-journée du fichier date.

Exemple : pour l'exemple considéré dans le fichier date, si l'on désire que la session pour une classe donnée débute le 1 juin au matin et se termine le 30 juin après-midi, alors

date de début de session = 47

date de fin de session = 106

car le 1 juin au matin est la 47<sup>ème</sup> demi-journée

le 30 juin après-midi est la 106<sup>ème</sup> demi-journée

par rapport aux demi-journées du fichier date.

Dans l'algorithme de coloration de graphe, cela voudra également dire en pratique que les séances d'examen pour cette classe donnée ne pourront prendre qu'une "couleur" comprise entre 47 et 106.

Les cours de cette classe sont des numéros qui font référence à un cours. On verra plus tard qu'un cours est identifié par un numéro.

### II.3.2. Les étudiants

=====

Le fichier correspondant aura la forme suivante :

Fichier étudiant dont chaque article se décompose de la façon suivante :

- 1) numéro d'étudiant
- 2) nom
- 3) prénom
- 4) examens pour lesquels il est inscrit

Ce fichier est organisé de façon séquentielle indexée. La clé d'accès d'un article étudiant est le numéro d'étudiant.

Le numéro d'étudiant est la concaténation de deux éléments

- 1) numéro de classe
- 2) numéro d'étudiant dans la classe.

Le numéro de classe fait référence à la classe dont le numéro est égale à ce numéro de classe.

Le numéro d'étudiant dans la classe est un numéro attribué par le programme à un étudiant de façon à ce que la clé **obtenue** soit identifiante.

Exemple : l'étudiant dont le numéro d'étudiant vaut 02104 peut être considéré comme le 104<sup>ème</sup> étudiant de la 2<sup>ème</sup> classe du fichier classe.

Le nom et le prénom d'un article étudiant sont repris pour permettre à un étudiant de se reconnaître et de reconnaître un étudiant.

Les examens pour lesquels un étudiant est inscrit sont représentés de la façon suivante :

1<sup>er</sup> niveau : ils sont représentés sous forme d'entiers positifs qui symbolisent en fait une suite de booléens (ou une suite de 0 ou 1)

2<sup>ème</sup> niveau : une fois la représentation décimale des examens traduite en représentation binaire, le résultat obtenu peut être considéré comme une suite de 0 et de 1.

Si dans cette suite, pour un étudiant donné (et donc, classe donnée), le  $i^{\text{ème}}$  élément de la suite vaut 0, alors cet étudiant n'est pas inscrit à l'examen portant sur le  $i^{\text{ème}}$  cours de la suite des cours de cette classe. Si cet élément vaut 1, alors cet étudiant est effectivement inscrit à l'examen portant sur le  $i^{\text{ème}}$  cours de la suite des cours de cette classe.

Remarque : le fait de s'être limité à 38 comme nombre maximum de cours pour une classe peut être quelque peu expliqué ici.

- 1) 38 semble un nombre entier suffisamment grand pour satisfaire la réalité pédagogique.
- 2) une suite de 38 0 ou 1 peut être représentée sous forme décimale par deux nombres entiers compris entre 0 et  $2^{38}$  (= 524288).



## II.3.3. Les professeurs

=====

Le fichier correspondant aura la forme suivante :

Fichier professeur dont chaque article se décompose de la façon suivante :

- 1) numéro de professeur  $i$       $1 \leq i \leq \text{nb-prof}$
- 2) nom
- 3) prénom
- 4) cours dont il est responsable ( $\text{val}(i)$   $1 \leq i \leq 8$ )
- 5) jour d'indisponibilité.

Ce fichier est organisé de façon relative. Le  $i^{\text{ème}}$  article de ce fichier est l'article dont le numéro de professeur =  $i$

Un professeur est identifié par son numéro de professeur.

Au départ, ce fichier est vide ( $\Rightarrow \text{nb-prof} = 0$ )

Le nom et le prénom permettent à un professeur de se reconnaître et de reconnaître un professeur.

Les cours dont un professeur est responsable sont représentés sous forme de numéro qui font référence à un cours. On verra plus tard qu'un cours est identifié par un numéro. Les jours d'indisponibilités d'un professeur sont représentés de la façon suivante :

1<sup>er</sup> niveau : ils sont représentés sous forme d'entiers positifs qui symbolisent en fait une suite de booléens (ou une suite de 0 ou 1)

2<sup>ème</sup> niveau : une fois ces entiers positifs traduits sous forme de nombres binaires, le résultat obtenu est équivalent à une suite de 0 et de 1.

Si dans cette suite, pour un professeur donné, le  $i^{\text{ème}}$  élément de la suite vaut 0, alors ce professeur est disponible pendant la  $i^{\text{ème}}$  demi-journée de l'ensemble des demi-journées de référence (cfr : fichier date). Si cet élément vaut 1, alors ce professeur est indisponible pendant cette demi-journée.

- Remarques : - le fait de limiter à 8 le nombre de cours par professeur peut s'expliquer par la réalité pédagogique.
- le fait que l'ensemble des demi-journées comprises par le fichier date représentent 114 demi-journées peut être quelque peu explicité ici.
- 1) 114 demi-journées semblent être un laps de temps suffisamment long pour reprendre un ensemble de session d'examen.
  - 2) une suite de 114 0 ou 1 peut être représentée sous forme décimale par 6 nombres entiers compris en 0 et  $2^{19} - 1$  c'est-à-dire six nombres compris entre 0 et 524287.

#### II.3.4. Les cours

=====

Le fichier correspondant aura la forme suivante

Fichier cours dont chaque article se décompose de la façon suivante :

- 1) numéro de cours  $i$   $1 \leq i \leq \text{nb-cours}$
- 2) intitulé
- 3) professeur
- 4) écrit
- 5) date de l'écrit
- 6) délai
- 7) min ou max
- 8) oral
- 9) nombre d'étudiants par oral
- 10) nombre de séances d'orale
- 11) date d'empêchement
- 12) samedi

Ce fichier est organisé de façon relative. Le  $i^{\text{ème}}$  article de ce fichier est l'article dont le numéro de cours =  $i$ .

Un cours est identifié par son numéro de cours.



Au départ, ce fichier est vide ( $\Rightarrow$  nb-cours = 0)

- L'intitulé (et le nom du professeur) permet(tent) de déterminer de façon précise de quel cours il s'agit.
- Le professeur responsable de ce cours est représenté sous forme du n° du professeur qui identifie le professeur. Cette information me servira lors de l'exécution de l'algorithme de coloration. En effet, toutes les demi-journées pendant lesquelles ce professeur est indisponible seront interdites à chaque séance d'examens (écrite ou orale) portant sur ce cours.
- L'écrit aura la valeur 0, 1 ou 2 suivant que pour ce cours il n'y a pas d'épreuve écrite ou qu'il y a une épreuve écrite durant une demi-journée ou une journée.
- La date de l'écrit (éventuel) aura une valeur  $1 \leq i \leq 114$  si l'écrit est fixé d'avance à la  $i^{\text{ème}}$  demi-journée de l'ensemble des demi-journées de référence. Si la date de l'écrit n'est pas fixée d'avance ou si il n'y a pas d'épreuve écrite pour ce cours, cette date vaut 0.
- Le délai représente un certain nombre de demi-journées qui doivent séparer l'épreuve orale de l'épreuve écrite. Ce délai peut être minimal (min = 0) si la première séance d'examen oral ne peut se dérouler qu'une fois ce délai passé après l'épreuve écrite (par exemple : la correction complète des copies de l'écrit prend deux jours et l'examen oral ne peut se dérouler que si les copies de l'écrit sont corrigées). Ce délai peut être maximal (min = 1) si la première séance d'examen oral doit se dérouler dans une certaine mesure juste après l'épreuve écrite (par exemple : un professeur désire faire passer l'écrit au matin et l'oral, l'après-midi). Il va de soit que le délai ne prend son sens que si l'examen d'un cours consiste en une épreuve écrite et une épreuve orale.
- Le nombre d'étudiants par orale et le nombre de scéances d'orale sont 2 notions complémentaires. En effet, si un examen portant sur un cours consiste notamment en une épreuve orale, le nombre de scéances orale pour ce cours sera déterminé de l'une de ces deux manières :
  - 1) ou bien ce nombre est fixé à l'avance. Dans ce cas, nombre de scéances d'orale =  $i \Rightarrow$  il y a  $i$  demi-journées consacrées à l'interrogation orale sur ce cours.



- 2) ou bien ce nombre est fixé suivant le nombre d'étudiants inscrits et le nombre d'étudiants qu'un professeur sait interroger en une demi-journée.
- Les dates d'empêchement représentent les dates pour lesquelles les scéances d'examen portant sur ce cours ne peuvent se dérouler. Elles sont représentées de la même façon que les jours d'indisponibilités d'un professeur. L'intérêt de cette notion est multiple.
- 1) intérêt pédagogique : Il est parfois souhaitable qu'un examen portant sur un cours se déroule avant un examen portant sur un autre cours (ex. : cours A 1<sup>ère</sup> partie et cours A 2<sup>ème</sup> partie). Dans ce cas, un moyen de formaliser ce desiderata est par exemple d'interdire l'examen sur le cours A<sub>1</sub> aux dates du 15 juin au 30 juin et d'interdire l'examen sur le cours A 2<sup>ème</sup> partie aux dates du 1 juin au 20 juin.
  - 2) Intérêt psychologique : Un professeur peut ordonnancer dans le temps le passage des examens portant sur ses cours, et ce dans une certaine mesure. Exemple : le professeur 1 est responsable des cours A et B. Il désire en avoir fini avec le cours A avant de commencer le cours B. Il lui suffira d'interdire la fin de session au cours A et d'interdire le début de session au cours B.
  - 3) Intérêt pratique : Si le programme d'horaire a donné une solution valable mais que l'on désire obtenir d'autres solutions, il suffira alors par exemple d'interdire la demi-journée j à la scéance s si la première solution attribuait la demi-journée j à la scéance s.
  - 4) Autre intérêt : Si on désire libérer une journée à une classe pour un motif ou l'autre, il suffira d'interdire cette journée à tous les cours de cette classe.
- le samedi : Suivant que les scéances d'examen portant sur ce cours peuvent ou ne peuvent pas se dérouler un samedi, samedi vaut 0 ou 1.



Remarques :

- 1) L'organisation des fichiers classes, cours et professeurs est relative car les références à un article classe, cours ou professeur se font par l'intermédiaire d'un numéro de classe, numéro de cours ou numéro de professeur. De là à rendre ces numéros clé d'accès relatif, il n'y avait qu'un pas. De plus, par construction de ces trois fichiers, si un fichier contient n articles, les clés d'accès de 1 à n font toutes références à un article de ce fichier et une clé d'accès > n ne fait référence à aucun article. Le fait d'utiliser des fichiers relatifs offre l'avantage par rapport au fichier séquentiel de modifier ou d'ajouter un article beaucoup plus facilement (dans le cadre de la saisie de donnée). Toutefois, dans le cadre du calcul de l'horaire, ces fichiers sont considérés comme équivalents à des fichiers séquentiels.
- 2) L'organisation du fichier étudiants est séquentielle indexée, dont la clé d'accès est une concaténation d'un n° de classe et d'un n° étudiant. en pratique, cela signifie que les étudiants d'une même classe sont " consécutifs " dans ce fichier, ce qui est un grand avantage. De plus, par rapport à un fichier d'organisation séquentielle, le fait d'ajouter, d'insérer ou de modifier un article est chose beaucoup plus aisée. Une organisation relative n'aurait pas été appropriée car la clé d'accès d'un article n'égale pas toujours la clé d'accès de l'article le précédant + 1, d'où perte de place (exemple : changement de classes, l'article n° 02104 précède l'article 03001  $\Rightarrow$  897 numéros non attribués). Ces considérations sont valables uniquement dans le cadre de la saisie des données. Dans le cadre même du calcul de l'horaire, ce fichier est considéré comme équivalent à un fichier séquentiel.
- 3) Les priorités que voici existent parmi les contraintes directes.
  - 1) date de l'écrit fixée : les examens sont placés aux dates fixées même si cela viole d'autres contraintes. Ces dates ne peuvent tomber un dimanche (pour des raisons évidentes) car contrôlées lors de la saisie des données.

- 2) dimanche, samedi, jours d'indisponibilités de professeur, délai
- 3) jours d'empêchement pour un cours
- 4) délai entre deux examens pour un étudiant
- 5) consécutivité des scéances orales d'un même examen.

Voici comment intervient cet ordre de priorité dans l'algorithme de coloration:

- les contraintes de priorité 1 et 2 doivent être obligatoirement respectées (à part qu'un examen dont la date est fixée d'avance peut violer toutes les autres contraintes, donc également les contraintes de priorité 2)
- Si la contrainte de priorité 3 rend la recherche d'une solution impossible, on rend cette contrainte moins forte (en rendant une demi-journée pour un cours c de nouveau disponible). Si cette contrainte ne peut plus être rendue moins forte, alors l'algorithme de coloration se termine sur un échec.
- Si la contrainte de priorité 4 rend la recherche d'une solution impossible, on décrémente le délai de 1. Si ce délai devient négatif, on remonte au niveau précédent et le délai d est réinitialisé à d-max.
- Si la contrainte de priorité 5 rend la recherche d'une solution impossible, alors on rend cette contrainte moins forte (en étalant les n scéances d'orale d'un même cours non plus sur k demi-journées mais sur  $k + 1$  demi-journées). Si cette contrainte ne peut plus être rendue moins forte, on remonte au niveau précédent et le temps de passage des orales d'un même cours est réinitialisé.



#### II.4. Coloration d'un sommet $i$ par une couleur $k$

=====

Soit un sommet  $i$  ( $1 \leq i \leq \text{nb-scea}$ )

une couleur  $k$  ( $1 \leq k \leq 114$ )

Si le processus de coloration attribue la couleur  $k$  au sommet  $i$ , alors  
si le sommet est le  $n^{\text{ième}}$  sommet coloré par le processus

$$f(i) = k \quad t\text{-or}(n) = k$$

$$L(i, k) = n$$

$$\text{npos}(i) = \text{npos}(i) - 1$$

De plus, il faut effectuer les éliminations pour tous les sommets adjacents au sommet  $i$ . Rappelons qu'un sommet  $j$  est adjacent à un sommet  $k$  pour trois raisons.

- 1) professeur en commun
- 2) étudiant en commun
- 3) cours en commun

Ces trois types d'adjacences vont provoquer le lancement de trois processus d'éliminations dans la matrice  $L$  afin que les contraintes soient respectées.

##### II.4.1. Sommets ayant professeur en commun

=====

Pour tous les sommet  $j$  ( $1 \leq j \leq \text{nb-scea}$ ) tel que le sommet  $j$  a le même professeur que le sommet  $i$ , il suffit d'interdire la couleur  $k$

Si  $f(j) = 0$  alors

si  $L(j, k) = 0$  alors  $L(j, k) = n$

$$\text{npos}(j) = \text{npos}(j) - 1$$

Remarques : 1)  $f(j) = 0$  pour ne faire les éliminations qu'aux sommets non encore coloriés.

2)  $L(j, k) = 0$  pour assurer que  $L(j, k)$  permet de retrouver le dernier sommet ayant provoqué cette élimination

#### II.4.2. Sommets ayant étudiant en commun

=====

Pour tous les sommets  $j$  ( $1 \leq j \leq \text{nb-scea}$ ) tel que  $t\text{-simul}(i, j) = 1$  (c-à-d. comprenant au moins 1 étudiant en commun avec le sommet  $i$ ), il faut interdire non seulement la couleur  $k$ , mais aussi un certain nombre de couleurs précédant ou suivant directement  $k$ . Notons que si ce nombre à la valeur  $d$ , ces éliminations assurent à chaque étudiant un délai de  $d$  demi-journées entre chaque examen.

Pour  $j$  tel que  $1 \leq j \leq \text{nb-scea}$ ,  $t\text{-simul}(i, j) = 1$  et  $f(j) = 0$   
 pour  $h$  allant de  $k - d$  jusque  $k + d$   
 si  $L(j, h) = 0$  alors  $L(j, h) = n$   
 $\text{npos}(j) = \text{npos}(j) - 1$

#### II.4.3. Sommets ayant cours en commun

=====

Supposons que le sommet  $i$  correspond au cours n°  $c$

Deux cas sont possibles :

- 1) le sommet  $i$  correspond à l'épreuve écrite du cours  $c$

soient  $i + 1, \dots, i + p$  ( $p \geq 0$ ) sommets correspondant aux épreuves orales du cours  $c$ .

Alors a) Il faut interdire à ces sommets toutes les couleurs allant de 1 à  $k$  (un écrit précédant un oral)

Pour  $j$  allant de  $i + 1$  à  $i + p$  tel que  $f(j) = 0$   
 pour  $h$  allant de 1 à  $k$   
 si  $L(j, h) = 0$  alors  $L(j, h) = n$   
 $\text{npos}(j) = \text{npos}(j) - 1$

- b) Si pour le cours  $c$ , un délai  $d$  minimum est exigé entre l'épreuve écrite et la première des séances orales, il faut interdire à ces sommets les couleurs allant de  $k + 1$  à  $k + d$ .



Pour  $j$  allant de  $i + 1$  à  $i + p$  tel que  $f(j) = 0$

Pour  $h$  allant de  $k + 1$  à  $k + d$

Si  $L(j, h) = 0$  alors  $L(j, h) = n$

$npos(j) = npos(j) - 1$

- c) Si pour le cours  $c$ , un délai  $d$  maximum est exigé entre l'épreuve écrite et la première des séances orales, il faut interdire à ces sommets les couleurs allant de  $k + d + p + 1$  jusque 114

Pour  $j$  allant de  $i + 1$  à  $i + p$  tel que  $f(j) = 0$

Pour  $h$  allant de  $k + d + p + 1$  jusque 114

Si  $L(j, h) = 0$  alors  $L(j, h) = n$

$npos(j) = npos(j) - 1$

- 2) le sommet  $i$  correspond à une séance d'orale du cours  $c$

=====

- a) Si le cours  $c$  comprend également une épreuve écrite, posons

$j$  = le sommet correspondant à l'épreuve écrite du cours  $c$

- 1) si pour le cours  $c$ , un délai  $d$  minimum est exigé entre l'épreuve écrite et la première des séances orales, il faut interdire au sommet  $j$  les couleurs allant de  $k - d$  à  $k$

si  $f(j) = 0$  alors

pour  $h$  allant de  $k - d$  à  $k$

si  $L(j, h) = 0$  alors  $L(j, h) = n$

$npos(j) = npos(j) - 1$

- 2) si pour le cours  $c$ , un délai  $d$  maximum est exigé entre l'épreuve écrite et la première des séances orales, il faut interdire au sommet  $j$  les couleurs allant de 1 à  $k - d - p - 1$  ( $p$  = nombre de séances orales portant sur le cours  $c$ )

si  $f(j) = 0$  alors

pour  $h$  allant de 1 à  $k - d - p - 1$

si  $L(j, h) = 0$  alors  $L(j, h) = n$

$npos(j) = npos(j) - 1$

- 3) interdire les couleurs allant de  $k$  à 114

pour  $h$  allant de  $k$  à 114

Si  $L(j, h) = 0$  alors  $L(j, h) = n$   $npos(j) = npos(j) - 1$

b) soient  $j + 1, \dots, j + p$  ( $p > 0$  et  $j + 1 \leq i \leq j + p$ ) les sommets correspondant aux épreuves orales du cours  $c$ . Pour assurer une certaine consécuité dans le temps de ces épreuves orales, il faut interdire à ces sommets les couleurs  $k$  tel que  $|h - k| \geq n\text{-max}$ . En procédant de cette façon, on assure que les  $p$  oraux du cours  $c$  se dérouleront dans un laps de temps inférieur à  $n\text{-max}$  demi-journées. Initialement,  $n\text{-max}$  a la valeur  $p$ .

Pour  $s$  allant de  $j + 1$  à  $j + p$  tel que  $f(s) = 0$   
 pour  $h$  allant de  $k - n\text{-max} - 1$  à  $k + n\text{-max} + 1$   
 si  $L(s, h) = 0$  alors  $L(s, h) = n$   
 $n\text{pos}(s) = n\text{pos}(s) - 1$

#### II.4.4. Conclusion

=====

Lorsque on attribue une couleur à un sommet, les éliminations effectuées dans la matrice de temps assurent que toutes les contraintes directes et indirectes seront respectées lors des colorations qui suivront.

#### II.5. L'implémentation

=====

Les algorithmes étudiés jusqu'ici ont été implémentés en Pascal sur l'ordinateur Dec-2060 et les conventions suivantes concernant les données du problème ont été prises

- Le nombre total des classes est limité à 50
- Le nombre des cours est limité à 500
- Le nombre des professeurs est limité à 500
- Pour chaque classe  $i$   $1 \leq i \leq 50$ 
  - le nombre d'étudiants est limité à 300
  - le nombre de cours est limité à 38
  - le nombre de séances d'exams est limité à 300
- La durée maximale de la session est de 114 demi-journées

Cela implique les matrices et variables maximales suivantes :

|                 |                         |
|-----------------|-------------------------|
| t-clas(1..50)   | matrice des classes     |
| t-cours(1..500) | matrice des cours       |
| t-prof(1..500)  | matrice des professeurs |
| t-étud(1..300)  | matrice des étudiants   |



c-c(1..38)        matrice des cours de la classe courante.  
t-simul(1..300,1..300) matrice de simultanéité des scéances  
L(1..300, 1...114)        matrice de temps  
f(1..300)        matrice des demi-journées.

Le processus de coloration se fait classe par classe, ce qui explique  
la taille des matrices    t-étud, c-c, t-simul, L, f.

C H A P I T R E      I I I



### Chapitre III : Le programme d'horaire d'examen =====

Un logiciel complet a été conçu de manière à ce qu'une personne non expérimentée en informatique puisse s'en servir facilement. Le logiciel comprend deux parties distinctes

- 1) gestion de fichier et saisie de données
- 2) calcul de l'horaire

Nous allons examiner la structure et fonctionnement pratique de ces deux parties

#### III.1. Gestion de fichier et saisie de données =====

9 | Cette partie est chargée de créer et de mettre à jour tous les fichiers nécessaires au calcul de l'horaire proprement dit. Cette partie a été écrite en cobol, langage de programmation des ordinateurs se voulant adapter à la résolution informatique des problèmes de type administratif. Nous allons décrire dans ce qui suit les différents modules composant cette partie gestion de fichier et saisie de données.

##### III.1.1. Module de gestion des fichiers =====

Pour chaque fichier existant, un module a été conçu pour assumer les primitives de mise à jour. Cela signifie notamment qu'une modification et d'accès de la structure interne d'un fichier (par exemple : par l'ajout d'items ne concernant en rien un logiciel de calcul d'horaire d'examens) n'implique que la modification du seul module assumant la gestion de ce fichier. Une étude complète des fichiers et de leur structure a été faite en II.3. Or, la description d'un type de données n'est complète que si à côté de leur structure, on spécifie les opérations permises sur les articles de chaque fichier. Nous allons pour chacun des modules de gestion de fichier énumérer la liste des paramètres et les opérations permises.

Ces différents modules sont appelés par des modules de niveaux supérieurs que nous verrons dans la suite. Les opérations permises varient selon le type de fichier et répondent aux besoins réels lors de la mise en place des fichiers du problème.

#### III.1.1.1. Gestion du fichier faculté = GEFACU

=====

##### a) liste des paramètres

- 1) T-op : type de l'opération
- 2) no : n° de l'article
- 3) article : article du fichier

NB : ce fichier est organisé de façon relative.

##### b) opération

- 1) T-op = "ini"

Fonction : lecture séquentielle de tous les articles du fichier et assignation de chaque article lu à un élément du tableau.  
(Ce tableau est une suite d'éléments dont la structure interne est identique à la structure d'un article du fichier). Le  $i^{\text{ème}}$  article du fichier est mis dans le  $i^{\text{ème}}$  élément du tableau.

Effet : Le fichier "faculté" a été mis dans un tableau mémoire centrale  
calcul de  $nf$  = nombre d'articles du fichier faculté.

- 2) T-op = "cpt"

Fonction : compter le nombre d'article du fichier

Effet :  $no = nf$

- 3) T-op = "cor"

Fonction : correspondance entre un numéro  $no$  et le  $no^{\text{ième}}$  article du  
fichier

Effet :  $article = tableau(no)$



## 4) T-op = "ecr"

Fonction : ajouter un nouvel article du fichier faculté

Effet :  $nf = nf + 1$

écriture à la  $nf^{ième}$  place du fichier de article

$tableau(nf) = article$

De plus, ce module est chargé de créer et de remplir un fichier Pfacu, fichier texte, c-à-d. accessible à des programmes écrits dans un langage de programmation quelconque. Cette opération sera effectuée lorsque l'opérateur aura fait savoir que la saisie de données est terminée et que le calcul de l'horaire proprement dit peut commencer.

## 5) T-op = "hor"

Fonction : création du fichier Pfacu

Effet : écriture dans le fichier Pfacu de tous les articles du fichier faculté.

## III.1.1.2. Gestion du fichier cycle=GECYCL

=====

a) liste des paramètres

1) T-op : type de l'opération

2) no : n° de l'article

3) article : article du fichier

NB : ce fichier est organisé de façon séquentielle

b) opération

1) T-op = "ini"

Fonction : lecture séquentielle de tous les articles du fichier et assignation de chaque article lu à un élément du tableau. (Ce tableau est une suite d'éléments dont la structure interne est identique à la structure d'un article du fichier). Le  $i^{ème}$  article du fichier est mis dans la  $i^{ème}$  élément du tableau.

Effet : Le fichier "cycle" a été mis en mémoire centrale  
calcul de  $nc = \text{nombre d'articles du fichier cycle.}$

2) T-op = "cpt"

Fonction : compter les articles du fichier = nc

Effet : no = nc

3) T-op = "cor"

Fonction : correspondance entre un numéro no et le no<sup>ième</sup> article du fichier

Effet : article = tableau(no)

De plus, ce module est chargé de créer un fichier Pcycl

4) T-op = "hor"

Fonction : création du fichier Pcycl

Effet : écriture dans le fichier Pcycl de tous les articles du fichier cycle.

#### III.1.1.3. Gestion du fichier date = Gedate

a) liste des paramètres

1) T-op : type d'opération

2) nb : nombre de 1 à 114

3) mois : valeur d'un mois

4) jour : valeur d'un jour

5) av/ap : avant ou après-midi

NB : ce fichier est organisé de façon séquentielle

b) T-op = "ini"

Fonction : lecture séquentielle de tous les articles du fichier et assignation de chaque article lu dans un tableau. (Ce tableau est une suite d'éléments dont la structure interne est identique à la structure d'un article du fichier). Le i<sup>ème</sup> article du fichier est mis dans le i<sup>ème</sup> élément du tableau.

Effet : Le fichier "date" a été mis dans un tableau en mémoire centrale.



## 2) T-op = "CDN"

Fonction : conversion d'une date (= mois, jour, av/ap) en un nombre de 1 à 114.

Effet : no = f (mois, jours, av/ap)

## 3) T-op = "CND"

Fonction : conversion d'un nombre de 1 à 114 en une date

Effet : mois =  $g_1(\text{no})$

jour =  $g_2(\text{no})$

av/ap =  $g_3(\text{no})$

De plus, ce module est chargé de créer un fichier Pdate

## 4) T-op = "hor"

Fonction : création du fichier Pdate

Effet : écriture dans le fichier Pdate de tous les articles du fichier "date"

## III.1.1.4. Gestion du fichier classe = g'eclas

=====

a) liste des paramètres

1) T-op : type d'opération

2) no : n° de l'article

3) id : identifiant d'une classe

4) EX/NEX : classe existant ou n'existant pas

5) article : article du fichier

NB. : fichier relatif

b) opérations

1) T-op = "ini"

Fonction : lecture séquentielle de tous les articles du fichier et assignation de chaque article lu à un élément d'un tableau (ce tableau est une suite d'éléments dont la structure interne est identique à la structure d'un article du fichier). Le  $i^{\text{ème}}$  article du fichier est mis dans le  $i^{\text{ème}}$  élément du tableau.

Effet : - Le fichier "classe" a été mis dans un tableau en mémoire centrale.

- calcul de  $nc$  = nombre d'articles du fichier classe.

2) T-op = "rec"

Fonction : étant donné id, rechercher un article du fichier dont  $id_{classe} = id$ .

Effet : si cet article existe et si c'est le  $i^{ème}$  alors

$EX/NEX = 1$

article = tableau (i)

sinon  $EX/NEX = 0$

3) T-op = "cor"

Fonction : correspondance entre un numéro  $no$  et le  $no^{ième}$  article du fichier.

Effet : article = tableau( $no$ )

4) T-op = "ecr"

Fonction : ajouter un nouvel article au fichier classe

Effet :  $nc = nc + 1$

Ecriture à la  $nc^{ième}$  place du fichier de article

tableau( $nc$ ) = article

5) T-op = "mod"

Fonction : modification d'un article du fichier classe

Effet : tableau( $no$ ) = article

réécriture à la  $no^{ième}$  place du fichier de article.

De plus, ce module est chargé de créer un fichier Pclas,...

6) T-op = "hor"

Fonction : création du fichier Pclas

Effet : écriture dans le fichier Pclas de tous les éléments du tableau.



## III.1.1.5. Gestion du fichier étudiant = Geetud

=====

a) liste des paramètres

- 1) T-op : type d'opération
  - 2) nocl : n° de classe
  - 3) noet : n° de l'étudiant dans la classe
  - 4) nom : nom d'un étudiant
  - 5) EX/NEX : étudiant existant ou n'existant pas
  - 6) article : article du fichier
- NB : fichier séquentiel indexé.

b) opération

- 1) T-op = "ini"

Fonction : lecture séquentielle de tous les articles du fichier dont le numéro de classe = nocl et assignation de ces articles à un élément d'un tableau. Le fichier étant séquentiel indexé, il suffit de donner comme valeur initiale de clé 1) numéro de classe = nocl et 2) numéro étudiant = 001 et de lire séquentiellement jusqu'à ce que l'article lu ait un numéro de classe  $\neq$  nocl

Effet : - tous les articles du fichier correspondant à la no<sup>ième</sup> classe ont été mis dans un tableau en mémoire centrale.  
- calcul de ne = nombre d'étudiants de cette classe.

- 2) T-op = "rec"

Fonction : étant donné un no de classe nocl et un nom, rechercher un article du fichier correspondant à la no<sup>ième</sup> classe et dont le nom = nom et dont le numéro d'étudiant noet

Effet : si cet article existe et si c'est le i<sup>ème</sup> article correspondant à la no<sup>ième</sup> classe alors

EX/NEX = 1

article = tableau(i)

sinon EX/NEX = 0

- 3) T-op = "cor"

Fonction : correspondance entre noet et nocl et le noet<sup>ième</sup> article de la classe nocl

Effet : article = tableau(no)

4) T-op = "cha"

Fonction : modifier le  $no^{ième}$  élément du tableau

Effet :  $tableau(no) = article$

5) T-op = "ecr"

Fonction : ajouter un nouvel étudiant à la classe noel

Effet :  $ne = ne + 1$

écriture de l'article dans le fichier

clé : numéro de classe = noel

numéro d'étudiant = ne

$tableau(ne) = article$

6) T-op = "mod"

Fonction : modification d'un article du fichier étudiant

Effet : réécriture dans le fichier de article

clé : numéro de la classe = noel

numéro de l'étudiant = noet

$tableau(no) = article$

7) T-op = "tec"

Fonction : modification de tous les articles du fichier étudiant

correspondant à la classe noel

Effet : Pour i allant de 1 à ne faire

réécriture dans le fichier de l'article dont la clé est tel que

- numéro de la classe = noel

- numéro de l'étudiant = i

par  $tableau(i)$

8) T-op = "sup"

Fonction : suppression d'un article du fichier étudiant

Effet : Pour i allant de noet jusqu'à  $ne - 1$

réécriture dans le fichier de l'article dont la clé est tel que

- numéro de la classe = noel

- numéro de l'étudiant = i

par  $tableau(i + 1)$ .

Suppression de l'article dont la clé est tel que

- numéro de la classe = noel

- numéro de l'étudiant = ne

$ne = ne - 1$



9) T-op = "hor"

Fonction : création du fichier Petud

Effet : écriture dans le fichier Pclas de tous les éléments du tableau.

III.1.1.6. Gestion du fichier professeur = Geprof

=====

a) liste des paramètres

1) T-op : type de l'opération

2) no : n° de l'article

3) nom : nom du professeur

4) EX/NEX : professeur existant ou n'existant pas

5) article : article du fichier

NB : ce fichier est organisé de façon relative

b) opérations

1) T-op = "ini"

Fonction : lecture séquentielle de tous les articles du fichier et assignation de chaque article lu à un élément du tableau. (Ce tableau est une suite d'éléments dont la structure interne est identique à la structure d'un article du fichier). Le  $i^{\text{ème}}$  article du fichier est mis dans le  $i^{\text{ème}}$  élément du tableau.

Effet : Le fichier professeur a été mis en mémoire centrale + calcul de np.

2) T-op = "rec"

Fonction : recherche d'un article du fichier dont le nom = nom et dont le numéro  $\geq$  no

Effet : si cet article existe et si c'est le  $i^{\text{ème}}$  alors

EX/NEX = 1

article = tableau(i)

sinon EX/NEX = 0

3) T-op = "cor"

Fonction : correspondance entre un numéro no et le  $no^{\text{ième}}$  article du fichier

Effet : article = tableau(no)

## 4) T-op = "ecr"

Fonction : ajouter un nouvel article au fichier professeur

Effet :  $np = np + 1$

écriture à la  $np^{ième}$  place du fichier de article

$tableau(np) = article$

## 5) T-op = "mod"

Fonction : modification d'un article du fichier professeur

Effet :  $tableau(no) = article$

réécriture à la  $no^{ième}$  place du fichier de article

De plus, ce module est chargé de créer un fichier Pprof...

## 6) T-op = "hor"

Fonction : création du fichier Pprof

Effet : écriture dans le fichier Pprof de tous les éléments du tableau

### III.1.1.7. Gestion du fichier cours = Gecour

=====

a) liste des paramètres

1) T-op : type d'opérations

2) no = n° de cours

3) article : article du fichier

NB : ce fichier est organisé de façon relative.

b) opérations

## 1) T-op = "ini"

Fonction : lecture séquentielle de tous les articles du fichier et assignation de chaque article lu à un élément d'un tableau (Ce tableau est une suite d'éléments dont la structure interne est identique à la structure d'un article du fichier). Le  $i^{ème}$  article du fichier est mis dans le  $i^{ème}$  élément du tableau.

Effet :-le fichier cours a été mis dans un tableau en mémoire centrale  
-calcul de  $nc$  = nombre d'articles du fichier cours

## 2) T-op = "cor"

Fonction : correspondance entre un numéro  $no$  et le  $no^{ième}$  article du fichier

Effet :  $article = tableau(no)$



3) T-op = "ecr"

Fonction : ajouter un nouvel article au fichier cours

Effet : nc = nc + 1

écriture à la nc<sup>ième</sup> place du fichier de article

tableau(nc) = article

4) T-op = "mod"

Fonction : modification d'un article du fichier cours

Effet : tableau(no) = article

réécriture à la no<sup>ième</sup> place du fichier de article

De plus, ce module est chargé de créer un fichier Pcour...

5) T-op = "hor"

Fonction : création du fichier Pcour

Effet : écriture dans le fichier Pcour de tous les éléments du tableau

### III.1.2. Modules intermédiaires

=====

Ces modules sont le résultat d'une démarche qui décompose chaque application en sous-applications. Ils sont appelés par un ou plusieurs modules applications et appellent eux-mêmes des modules de gestion de fichiers. Nous allons brièvement pour chacun de ces modules définir les paramètres et les applications effectuées.

#### III.1.2.1. Identification d'un professeur = idprof

=====

a) liste des paramètres

1) nom

2) prénom

3) art-exis

4) no article

b) fonction du module

1) Saisir le nom et le prénom d'un professeur.

- 2) vérifier si ces données correspondent à un article du fichier professeur
  - si oui art-exis = 1
  - no article = numéro de cet article
  - si non art-exis = 0

III.1.2.2. Identification d'un étudiant = idetud  
=====

a) liste des paramètres

- 1) nom
- 2) prénom
- 3) art-exis
- 4) no-article

b) fonction du module

- 1) saisir le nom et le prénom d'un étudiant
- 2) vérifier si ces données correspondent à un article du fichier étudiant
  - si oui art-exis = 1
  - no article = numéro de cet article
  - si non art-exis = 0

III.1.2.3. Identification d'une classe = idclas  
=====

a) liste des paramètres

- 1) nofac : n° de faculté de la classe
- 2) nocyc : n° de cycle de la classe
- 3) année : année de la classe
- 4) art-exis
- 5) no-article

b) fonctions du module

- 1) saisir la faculté, le cycle et l'année d'une classe
- 2) vérifier si ces données correspondent à un article du fichier classe
  - si oui art-exis = 1
  - no article = numéro de cet article
  - si non art-exis = 0



#### III.1.2.4. Modification d'un cours = mocour =====

- a) liste des paramètres
  - 1) nocl : n° de la classe
  - 2) noco : n° de cours
  - 3) art-exis
- b) fonctions
  - 1) modifier l'intitulé du cours
  - 2) modifier le professeur responsable du cours
  - 3) supprimer ce cours de la liste des cours d'une classe  
dans ce cas art-exis = 1 sinon art-exis = 0

#### III.1.2.5. Saisie d'un cours = sacour =====

- a) liste des paramètres
  - 1) nocl : n° de la classe
  - 2) noco : n° du cours
- b) fonctions
  - 1) saisir l'intitulé et le professeur responsable de ce cours

#### III.1.3. Modules application =====

Chacun de ces modules implémentent une application de la saisie de données. Nous allons, pour chacun de ces modules, reprendre brièvement la liste des fonctions effectuées.

#### III.1.3.4. Module saisie des cours = sacour =====

- 1) saisie, modification, affichage ou impression de la liste des cours d'une classe.

#### III. 1.3.5. Module saisie des étudiants = saetud =====

- 1) saisie, modification, affichage ou impression de la liste des étudiants d'une classe.

III.1.3.3. Module saisie des professeurs = saprof  
=====

- 1) saisie, modification, affichage ou impression de la liste des professeurs

III. 1.3.4. Module des modalités d'examen = samoda  
=====

- 1) saisie des modalités d'examen d'un cours

III.1.3.5. Module inscription aux examens = sainx  
=====

- 1) saisie, affichage ou impression de la liste des examens auxquels un étudiant est inscrit

III.1.3.6. Module indisponibilité d'un professeur = saindi  
=====

- 1) saisie, impression ou affichage des jours d'indisponibilité d'un professeur

III.1.3.7. Module date de la session = sadate  
=====

- 1) saisie, affichage ou impression des dates de début et de fin de session d'une classe

III.1.4. Modules menus  
=====

III.1.4.1. Module menu des saisies = menusa  
=====

- 1) saisie d'une des 7 applications proposées ou retour au PP



### III.1.4.2. Module directeur = PP

=====

- 1) initialisation de toutes les données
- 2) Proposition à l'utilisateur de 3 possibilités
  - 1) saisie de données
  - 2) calcul de l'horaire
  - 3) arrêt

Si 1) appel de Menusa

- 2) 1) Détermination des classes pour lesquels l'horaire doit être calculé
- 2) écriture des fichiers textes qui sont le trait d'union entre les deux parties
- 3) fin de l'exécution

### III.2. Calcul de l'horaire

=====

Cette partie est chargée de calculer un horaire d'examen à partir de tous les fichiers créés par la partie gestion de fichier et saisie de données et d'imprimer les résultats. Cette partie a été écrite en Pascal, langage de programmation des ordinateurs adapté aux applications scientifiques.

L'impression des résultats se fait

- 1) classe par classe
- 2) demi-journée par demi-journée
- 3) cours par cours
- 4) heure par heure
- 5) étudiants par étudiants

#### Exemple

#### Première licence informatique

|                    |   |
|--------------------|---|
| Le 2 juin au matin |   |
| Analyse organique  |   |
| 8 h 30             | A |
|                    | B |
| 9 h 30             | C |
| :                  | : |
| Cobol              | : |
| 8 h 30             | D |
| :                  | : |
| Le 3 juin au matin | : |
| :                  | : |
| :                  | : |

Deuxième licence informatique

Le 2 juin au matin

⋮

Cette partie consiste en une seule unité compilable et ne demande qu'une interaction limitée de la part de l'utilisateur.



C H A P I T R E      I V

## Chapitre IV : Expériences pratiques

=====

Le programme d'horaire d'examen dont la structure est décrite au troisièm<sup>e</sup> chapitre, a été implémenté et mis sous forme de "software package".

Les tests effectués dans l'institut informatique, les conclusions pratiques correspondantes sont décrits dans ce chapitre.

De plus, quelques résultats ont été mis pour l'exemple.

### IV.1. Tests effectués

=====

Les tests ont été effectués sur la session de juin 1983 dans l'institut d'informatique (licence et maîtrise). Le travail d'établissement d'un horaire comprend trois grandes parties.

#### 1) Saisie des données de l'horaire

Etablir pour chaque classe

- 1) la liste des cours et des étudiants + la date de début et fin de session

Etablir pour chaque cours

- 1) les modalités de l'examen

Etablir pour chaque professeur

- 1) la liste des demi-journées d'indisponibilité

Etablir pour chaque étudiant

- 1) la liste des examens auxquels il est inscrit

Ce travail de secrétariat est fondamental pour que l'horaire réponde bien aux exigences réelles. Il est très important d'entrer toutes les informations disponibles pour que l'ordinateur puisse au mieux tenir compte de la situation présente vu "qu'il ne faut jamais supposer que l'ordinateur suppose quelque chose".



2) Encodage des données

Les données préparées en 1) sont entrées par terminal, ce qui se fait très facilement après lecture du manuel utilisateur (voir en annexe)

3) Calcul de l'horaire

Une fois que toutes les données ont été mises sur fichier, on procède au calcul de l'horaire proprement dit. Il est parfois utile de travailler sur des données différentes (en jouant par exemple sur les jours d'empêchement d'un cours) pour obtenir des horaires différents. Il appartiendra de comparer la "qualité" de ces différents horaires et, selon des critères bien évidemment subjectifs, de choisir parmi eux, celui qui sera appliqué. A noter qu'un des ces critères, (c-à-d. le délai minimum pour chaque étudiant entre deux examens) est donné par le programme, la logique voulant que l'on maximalise ce délai minimum). Ce critère n'est toutefois sûrement pas absolu.

L'horaire d'examen pour les deux premières années de l'institut d'informatique a été calculé pour la session de juin 1983. Les tests furent très satisfaisants et mettent en évidence la fonctionabilité du logiciel.

IV.4. Résultats

Dans les pages qui se suivent, se trouvent un exemple d'horaire pour la première licence et maîtrise informatique.



\*\*\*\*\*  
 \*HORAIRE D'EXAMENS DE LA 1 LICENCE ET MAITRISE INFORMATIQUE  
 \*\*\*\*\*

\*\*\*\*\*  
 \* LE 1 JUIN AU MATIN \*  
 \*\*\*\*\*

COURS DE ECONOMIE DES ORGANISATIONS  
 EPREUVE ECRITE

GUILLAUME

COURS DE INTR. A L'ECONOMIE DES ORGANIS

SCHEPENS

|              |           |            |
|--------------|-----------|------------|
| 8 heures 30  | WILLIQUET | MARC       |
|              | AGUILAR   | VICTOR     |
| 9 heures 30  | BAUDOUX   | GUY        |
|              | PICHOT    | MICHEL     |
|              | MARTIN    | VINCENT    |
| 10 heures 30 | MOULART   | PASCAL     |
|              | TRIGEAUX  | BENOIT     |
|              | WILMES    | HORST      |
| 11 heures 30 | HENDRICK  | ANNE       |
|              | JACQUET   | JEAN-MARIE |
|              | LORGE     | FREDDY     |
|              | MULLER    | DENISE     |

\*\*\*\*\*  
 \* LE 1 JUIN APRES-MIDI \*  
 \*\*\*\*\*

COURS DE INTR. A L'ECONOMIE DES ORGANIS

SCHEPENS

|           |              |         |
|-----------|--------------|---------|
| 14 heures | GUIOT        | DIDIER  |
|           | TERRYN       | PASCALE |
| 15 heures | TOURNAY      | PATRICK |
|           | GUILLAUME    | GILBERT |
|           | HOANG        | HOA     |
| 16 heures | VANHEMELRYCK | CLAUDE  |
|           | GILLIN       | PATRICK |
|           | MOONS        | XAVIER  |
| 17 heures | PIERRE       | BRUNO   |
|           | VANIMPE      | M-FR    |
|           | DETAILLE     | J.-CL   |
|           | LEFEBVRE     | PASCALE |

\*\*\*\*\*  
 \* LE 2 JUIN AU MATIN \*  
 \*\*\*\*\*

COURS DE THEORIE DES ORGANISATIONS

DRABS

|              |           |              |
|--------------|-----------|--------------|
| 8 heures 30  | DUCARME   | PHILIPPE     |
|              | VANDEPUT  | MICHEL       |
|              | MULLER    | JOSIANE      |
| 9 heures 30  | WARZEE    | PIERRE       |
|              | GUILLAUME | GILBERT      |
|              | HALLO     | MARIA        |
| 10 heures 30 | WATERKEYN | PASCAL       |
|              | JOSIS     | ALAIN        |
|              | DANGIS    | JEAN-MICHEL  |
| 11 heures 30 | DE SWERT  | PIERRES-YVES |
|              | RYS       | XAVIER       |
|              | BOUKHCHA  | SAID         |
|              | DINANT    | JEAN-MARC    |



## COURS DE INTR. A L'ECONOMIE DES ORGANIS

SCHEPENS

|              |                               |                                   |
|--------------|-------------------------------|-----------------------------------|
| 8 heures 30  | VAUSE<br>DEVAUX<br>DELAITE    | MICHEL<br>YVES<br>YVES            |
| 9 heures 30  | DEBOIS<br>HENDE<br>VAN DOORNE | J.-J.<br>KONGA<br>JEAN-LOUIS      |
| 10 heures 30 | HUCQ<br>CLANTIN<br>DUBUISSON  | BENOIT<br>MARCEL<br>PIERRE        |
| 11 heures 30 | ELIAS<br>AUBERT<br>MORAUX     | CHRISTIAN<br>BERNARD<br>FRANCOISE |

\*\*\*\*\*  
\* LE 3 JUIN AU MATIN \*  
\*\*\*\*\*

## COURS DE INFORMATIQUE ET SCIENCES

BERLEUR

|             |                               |                             |
|-------------|-------------------------------|-----------------------------|
| 8 heures 30 | MULLER<br>WARZEE<br>WATERKEYN | JOSIANE<br>PIERRE<br>PASCAL |
| 9 heures 30 | JOSIS<br>DANGIS               | ALAIN<br>JEAN-MICHEL        |

|              |                    |                        |
|--------------|--------------------|------------------------|
| 10 heures 30 | DE SWERT<br>RYS    | PIERRES-YVES<br>XAVIER |
| 11 heures 30 | BOUKHCHA<br>DINANT | SAID<br>JEAN-MARC      |

\*\*\*\*\*  
\* LE 3 JUIN APRES-MIDI \*  
\*\*\*\*\*

## COURS DE INTR. A L'ECONOMIE DES ORGANIS

SCHEPENS

|           |                               |                                |
|-----------|-------------------------------|--------------------------------|
| 14 heures | MEURISSE<br>VALENTIN<br>STAES | MYRIAM<br>DIDIER<br>OLIVIER    |
| 15 heures | CANART<br>MAREE<br>HAZARD     | JEAN-YVES<br>JACQUES<br>PASCAL |
| 16 heures | DAHMEN<br>VAN GAEVER<br>CAPON | GUY<br>RUDY<br>FRANCOIS        |
| 17 heures | EVARD<br>WENZI                | MARC<br>MENAYANE               |

\*\*\*\*\*  
\* LE 4 JUIN AU MATIN \*  
\*\*\*\*\*

## COURS DE ELEMENTS D'EVAL. D'ARCHITECTUR

BRUNIN

EPREUVE ECRITE



\*\*\*\*\*  
 \* LE 6 JUIN APRES-MIDI \*  
 \*\*\*\*\*

COURS DE ELEMENTS D'EVAL. D'ARCHITECTUR BRUNIN

|           |                                           |                                      |
|-----------|-------------------------------------------|--------------------------------------|
| 14 heures | WILLIQUET<br>AGUILAR<br>SAUDOUX<br>PICHOT | MARC<br>VICTOR<br>GUY<br>MICHEL      |
| 15 heures | MARTIN<br>MOULART<br>TRIGEAUX<br>NILMES   | VINCENT<br>PASCAL<br>BENOIT<br>HORST |
| 16 heures | MENDRICK                                  | ANNE                                 |

|           |                                                        |                                                   |
|-----------|--------------------------------------------------------|---------------------------------------------------|
| 17 heures | JACQUET<br>LORGE<br>MULLER<br>VERRIEST<br>VANGEERSDAEL | JEAN-MARIE<br>FREDDY<br>DENISE<br>ALAIN<br>JOELLE |
|-----------|--------------------------------------------------------|---------------------------------------------------|

\*\*\*\*\*  
 \* LE 7 JUIN AU MATIN \*  
 \*\*\*\*\*

COURS DE ELEMENTS D'EVAL. D'ARCHITECTUR BRUNIN

|              |                                            |                                      |
|--------------|--------------------------------------------|--------------------------------------|
| 8 heures 30  | TERRYN<br>TOURNAY<br>GUILLAUME<br>HOANG    | PASCALE<br>PATRICK<br>GILBERT<br>HOA |
| 9 heures 30  | VANHEMELDRYCK<br>GILLIN<br>MOONS<br>PIERRE | CLAUDE<br>PATRICK<br>XAVIER<br>BRUNO |
| 10 heures 30 | VANIMPE<br>DETAILLE<br>LEFEBVRE            | M-FR<br>J.-CL<br>PASCALE             |
| 11 heures 30 | CATHY<br>LE KHAM<br>VLIETINCK              | J.M.<br>MICHEL                       |

COURS DE ELEMENTS D'ARCHITECTURE CHERTON

|              |                    |                       |
|--------------|--------------------|-----------------------|
| 8 heures 30  | WENZEL<br>ROTENS   | MENAYANE<br>ETIENNE   |
| 9 heures 30  | RIFFONT<br>DUCARME | CHRISTINE<br>PHILIPPE |
| 10 heures 30 | VANDEPUT<br>WARZEE | MICHEL<br>PIERRE      |
| 11 heures 30 | HALLO<br>WATERKEYN | MARIA<br>PASCAL       |

COURS DE INFORMATIQUE ET SCIENCES BERLEUR

|              |                               |                           |
|--------------|-------------------------------|---------------------------|
| 8 heures 30  | OGER<br>PIREN<br>CADELLI      | LAURA<br>SYLVAIN<br>MARIO |
| 9 heures 30  | BURNAY<br>BOUCHAT<br>MEURISSE | JEAN<br>PIERRE<br>MYRIAM  |
| 10 heures 30 | VALENTIN<br>STAES             | DIDIER<br>OLIVIER         |
| 11 heures 30 | CANART                        | JEAN-YVES                 |

BERNARD

JEAN-MARIE



\*\*\*\*\*  
 \* LE 7 JUIN APRES-MIDI \*  
 \*\*\*\*\*

COURS DE ELEMENTS D'EVAL. D'ARCHITECTUR

BRUNIN

|           |                                        |                                           |
|-----------|----------------------------------------|-------------------------------------------|
| 14 heures | LEBEGUE<br>TASIAUX<br>GRAAS<br>CARLIER | HENRY<br>BRIGITTE<br>ROBERT<br>JEAN-LOUIS |
| 15 heures | LEROY<br>LABRANCHE<br>RENAULD<br>VAUSE | ANDRE<br>MARC<br>YVAN<br>MICHEL           |
| 16 heures | DEVAUX<br>DELAITE<br>DEBOIS            | YVES<br>YVES<br>J.-J.                     |
| 17 heures | MENDE<br>VAN DOORNE<br>HUCQ            | KONGA<br>JEAN-LOUIS<br>BENOIT             |

COURS DE ELEMENTS D'ARCHITECTURE

CHERTON

|           |                      |                      |
|-----------|----------------------|----------------------|
| 14 heures | WILLIQUET<br>AGUILAR | MARC<br>VICTOR       |
| 15 heures | BAUDOUX<br>PICHOT    | GUY<br>MICHEL        |
| 16 heures | WILMES<br>HENDRICK   | HORST<br>ANNE        |
| 17 heures | JACQUET<br>LORGE     | JEAN-MARIE<br>FREDDY |

\*\*\*\*\*  
 \* LE 8 JUIN AU MATIN \*  
 \*\*\*\*\*

COURS DE THEORIE DES ORGANISATIONS

DRABS

|             |                                         |                                                     |
|-------------|-----------------------------------------|-----------------------------------------------------|
| 8 heures 30 | BERNARD<br>MAREE<br>DE PAUW<br>JUSSERET | JEAN-MARIE<br>JACQUES<br>FABIENNE<br>MARI-CHRISTINE |
| 9 heures 30 | HAZARD<br>DAHMEN<br>VAN GAEVER          | PASCAL<br>GUY<br>RUDY                               |

|              |                            |                                  |
|--------------|----------------------------|----------------------------------|
| 10 heures 30 | CAPON<br>LIEPIN<br>EVRARD  | FRANCOIS<br>SERGE<br>MARC        |
| 11 heures 30 | WENZI<br>ROTEUS<br>RIFFONT | MENAYANE<br>ETIENNE<br>CHRISTINE |

COURS DE ELEMENTS D'EVAL. D'ARCHITECTUR

BRUNIN

|              |                                         |                                          |
|--------------|-----------------------------------------|------------------------------------------|
| 8 heures 30  | CLANTIN<br>DUBUISSON<br>ELIAS<br>AUBERT | MARCEL<br>PIERRE<br>CHRISTIAN<br>BERNARD |
| 9 heures 30  | MORAU<br>BEYNE-MAIRIAUX<br>PIERRE       | FRANCOISE<br>FR.<br>BERNARD              |
| 10 heures 30 | VANHENTENRYCK<br>JEANJOT<br>DEMOULIN    | PASCAL<br>BRUNO<br>JEAN-PIERRE           |
| 11 heures 30 | KAYSER<br>VISART<br>MANISE              | PASCAL<br>MANOEL<br>PATRICK              |



## COURS DE ELEMENTS D'ARCHITECTURE

CHERTON

|              |              |         |
|--------------|--------------|---------|
| 8 heures 30  | VERRIEST     | ALAIN   |
|              | VANGEERSDAEL | JOELLE  |
| 9 heures 30  | GUIOT        | DIDIER  |
|              | TERRYN       | PASCALE |
| 10 heures 30 | TOURNAY      | PATRICK |
|              | GUILLAUME    | GILBERT |
| 11 heures 30 | GILLIN       | PATRICK |
|              | MOONS        | XAVIER  |

\*\*\*\*\*  
 \* LE 8 JUIN APRES-MIDI \*  
 \*\*\*\*\*

## COURS DE ELEMENTS D'EVAL. D'ARCHITECTUR

BRUNIN

|           |          |          |
|-----------|----------|----------|
| 14 heures | CAVEZ    | J.J.     |
|           | SQUILBIN | J.M.     |
|           | DHEUR    | EMMANUEL |
|           | WEICKER  | MICHEL   |
| 15 heures | OGER     | LAURA    |
|           | PIREN    | SYLVAIN  |
|           | CADELLI  | MARIO    |
| 16 heures | BURNAY   | JEAN     |
|           | BOUCHAT  | PIERRE   |

|           |          |           |
|-----------|----------|-----------|
| 17 heures | MEURISSE | MYRIAM    |
|           | VALENTIN | DIDIER    |
|           | STAES    | OLIVIER   |
|           | CANART   | JEAN-YVES |

## COURS DE ELEMENTS D'ARCHITECTURE

CHERTON

|           |           |            |
|-----------|-----------|------------|
| 14 heures | DETAILLE  | J.-CL      |
|           | LEFEBVRE  | PASCALE    |
| 15 heures | GATHY     | J.M.       |
|           | LE KHAM   | MICHEL     |
| 16 heures | VLIETINCK | BRIGITTE   |
|           | TASIAUX   | JEAN-LOUIS |
| 17 heures | CARLIER   | ANDRE      |
|           | LEROY     |            |

\*\*\*\*\*  
 \* LE 9 JUIN AU MATIN \*  
 \*\*\*\*\*

## COURS DE THEORIE DES ORGANISATIONS

DRABS

|              |              |            |
|--------------|--------------|------------|
| 8 heures 30  | WILLIQUET    | MARC       |
|              | AGUILAR      | VICTOR     |
|              | BAUDOUX      | GUY        |
|              | PICHOT       | MICHEL     |
| 9 heures 30  | MARTIN       | VINCENT    |
|              | MOULART      | PASCAL     |
|              | TRIGEAUX     | BENOIT     |
|              | WILMES       | HORST      |
| 10 heures 30 | HENDRICK     | ANNE       |
|              | JACQUET      | JEAN-MARIE |
|              | LORGE        | FREDDY     |
| 11 heures 30 | MULLER       | DENISE     |
|              | VERRIEST     | ALAIN      |
|              | VANGEERSDAEL | JOELLE     |



## COURS DE ELEMENTS D'EVAL. D'ARCHITECTUR

BRUNIN

|              |                                         |                                                     |
|--------------|-----------------------------------------|-----------------------------------------------------|
| 8 heures 30  | BERNARD<br>MAREE<br>DE PAUW<br>JUSSERET | JEAN-MARIE<br>JACQUES<br>FABIENNE<br>MARI-CHRISTINE |
| 9 heures 30  | HAZARD<br>DAHMEN<br>VAN GAEVER          | PASCAL<br>GUY<br>RUDY                               |
| 10 heures 30 | CAPON<br>LIEPIN<br>EVRARD               | FRANCOIS<br>SERGE<br>MARC                           |

|              |                            |                                  |
|--------------|----------------------------|----------------------------------|
| 11 heures 30 | WENZI<br>ROTENS<br>RIFFONT | MENAYANE<br>ETIENNE<br>CHRISTINE |
|--------------|----------------------------|----------------------------------|

## COURS DE ELEMENTS D'ARCHITECTURE

CHERTON

|              |                    |                      |
|--------------|--------------------|----------------------|
| 8 heures 30  | RENAULD<br>VAUSE   | YVAN<br>MICHEL       |
| 9 heures 30  | DEVAUX<br>DELAITE  | YVES<br>YVES         |
| 10 heures 30 | DEBOIS<br>MENDE    | J.-J.<br>KONGA       |
| 11 heures 30 | VAN DOORNE<br>HUCQ | JEAN-LOUIS<br>BENOIT |

\*\*\*\*\*  
 \* LE 9 JUIN APRES-MIDI \*  
 \*\*\*\*\*

## COURS DE THEORIE DES ORGANISATIONS

DRABS

|           |                                          |                                         |
|-----------|------------------------------------------|-----------------------------------------|
| 14 heures | GUIOT<br>TERRYN<br>TOURNAY<br>GUILLAUME  | DIDIER<br>PASCALE<br>PATRICK<br>GILBERT |
| 15 heures | HOANG<br>VANHEMELRYCK<br>GILLIN          | HOA<br>CLAUDE<br>PATRICK                |
| 16 heures | MOONS<br>PIERRE<br>VANIMPE               | XAVIER<br>BRUNO<br>M-FR                 |
| 17 heures | DETAILLE<br>LEFEBVRE<br>GATHY<br>LE KHAM | J.-CL<br>PASCALE<br>J.M.                |

## COURS DE ELEMENTS D'EVAL. D'ARCHITECTUR

BRUNIN

|           |                                         |                                         |
|-----------|-----------------------------------------|-----------------------------------------|
| 14 heures | DUCARME<br>VANDEPUT<br>MULLER<br>WARZEE | PHILIPPE<br>MICHEL<br>JOSIANE<br>PIERRE |
| 15 heures | GUILLAUME<br>HALLO<br>WATERKEYN         | GILBERT<br>MARIA<br>PASCAL              |
| 16 heures | JOSIS<br>DANGIS<br>DE SWERT             | ALAIN<br>JEAN-MICHEL<br>PIERRES-YVES    |
| 17 heures | RYS                                     | XAVIER                                  |

BOUKHCHA  
DINANTSAID  
JEAN-MARC



## COURS DE ELEMENTS D'ARCHITECTURE

CHERTON

|           |                |          |
|-----------|----------------|----------|
| 14 heures | BEYNE-MAIRIAUX | FR.      |
| 15 heures | VANHENTENRYCK  | PASCAL   |
|           | VISART         | MANOEL   |
| 16 heures | MANISE         | PATRICK  |
|           | CAVEZ          | J.J.     |
| 17 heures | SQUILBIN       | J.M.     |
|           | DHEUR          | EMMANUEL |
|           | WEICKER        | MICHEL   |

\*\*\*\*\*  
\* LE 10 JUIN AU MATIN \*  
\*\*\*\*\*

## COURS DE THEOPIE DES ORGANISATIONS

DRABS

|              |            |            |
|--------------|------------|------------|
| 8 heures 30  | VLIETINCK  | MICHEL     |
|              | LEBEGUE    | HENRY      |
|              | TASIAUX    | BRIGITTE   |
|              | GRAAS      | ROBERT     |
| 9 heures 30  | CARLIER    | JEAN-LOUIS |
|              | LEROY      | ANDRE      |
|              | LABRANCHE  | MARC       |
|              | RENAULD    | YVAN       |
| 10 heures 30 | VAUSE      | MICHEL     |
|              | DEVAUX     | YVES       |
|              | DELAITE    | YVES       |
| 11 heures 30 | DEBOIS     | J.-J.      |
|              | MENDE      | KONGA      |
|              | VAN DOORNE | JEAN-LOUIS |

\*\*\*\*\*  
\* LE 11 JUIN AU MATIN \*  
\*\*\*\*\*

## COURS DE ANALYSE FONCTIONNELLE

BODART

## EPREUVE ECRITE

\*\*\*\*\*  
\* LE 13 JUIN AU MATIN \*  
\*\*\*\*\*

## COURS DE TECHNOLOGIE DES FICHIERS

HAINAUT

## EPREUVE ECRITE



\*\*\*\*\*  
 \* LE 14 JUIN AU MATIN \*  
 \*\*\*\*\*

COURS DE THEORIE DES GRAPHS  
 EPREUVE ECRITE

FICHEFET

\*\*\*\*\*  
 \* LE 15 JUIN AU MATIN \*  
 \*\*\*\*\*

COURS DE THEORIE DES GRAPHS

FICHEFET

|              |           |            |
|--------------|-----------|------------|
| 8 heures 30  | WILLIQUET | MARC       |
|              | AGUILAR   | VICTOR     |
| 9 heures 30  | BAUDOUX   | GUY        |
|              | PICHOT    | MICHEL     |
|              | MARTIN    | VINCENT    |
| 10 heures 30 | MOULART   | PASCAL     |
|              | TRIGEAUX  | BENOIT     |
|              | WILMES    | HORST      |
| 11 heures 30 | HENDRICK  | ANNE       |
|              | JACQUET   | JEAN-MARIE |
|              | LORGE     | FREDDY     |
|              | MULLER    | DENISE     |

COURS DE PROCESSUS STOCHASTIQUE

NOIRHOMME

|              |              |         |
|--------------|--------------|---------|
| 8 heures 30  | VERRIEST     | ALAIN   |
|              | VANGEERSDAEL | JOELLE  |
| 9 heures 30  | TERRYN       | PASCALE |
|              | TOURNAY      | PATRICK |
| 10 heures 30 | GUILLAUME    | GILBERT |
|              | HOANG        | HOA     |
| 11 heures 30 | VANHEMELRYCK | CLAUDE  |
|              | PIERRE       | BRUNO   |
|              | VANIMPE      | M-FR    |

COURS DE THEORIE DES ORGANISATIONS

DRABS

|              |          |           |
|--------------|----------|-----------|
| 8 heures 30  | CAVEZ    | J.J.      |
|              | SQUILBIN | J.M.      |
|              | DHEUR    | EMMANUEL  |
| 9 heures 30  | WEICKER  | MICHEL    |
|              | OGER     | LAURA     |
|              | PIREN    | SYLVAIN   |
| 10 heures 30 | CADELLI  | MARIO     |
|              | BURNAY   | JEAN      |
|              | BOUCHAT  | PIERRE    |
| 11 heures 30 | MEURISSE | MYRIAM    |
|              | VALENTIN | DIDIER    |
|              | STAES    | OLIVIER   |
|              | CANART   | JEAN-YVES |

COURS DE SYSTEMES D'EXPLOITATIONS

RAMAEKERS

|              |            |            |
|--------------|------------|------------|
| 8 heures 30  | VAUSE      | MICHEL     |
|              | DEVAUX     | YVES       |
| 9 heures 30  | DELAITE    | YVES       |
|              | DEBOIS     | J.-J.      |
|              | MENDE      | KONGA      |
| 10 heures 30 | VAN DOORNE | JEAN-LOUIS |
|              | HUCQ       | BENOIT     |
|              | CLANTIN    | MARCEL     |
| 11 heures 30 | DURUISSON  | PIERRE     |
|              | ELIAS      | CHRISTIAN  |
|              | AUBERT     | BERNARD    |



\*\*\*\*\*  
 \* LE 15 JUIN APRES-MIDI \*  
 \*\*\*\*\*

COURS DE THEORIE DES GRAPHS

FICHEFET

|           |                                 |                               |
|-----------|---------------------------------|-------------------------------|
| 14 heures | DETAILLE<br>LEFEBVRE<br>GATHY   | J.-CL<br>PASCALE<br>J.M.      |
| 15 heures | LE KHAM<br>VILETINCK<br>LIBEGUE | MICHEL<br>HENRY<br>BRIGITTE   |
| 16 heures | TASIAUX<br>GRAAS<br>CARLIER     | ROBERT<br>JEAN-LOUIS<br>ANDRE |
| 17 heures | LEROY<br>LABRANCHE<br>RENAULD   | MARC<br>YVAN                  |

COURS DE PROCESSUS STOCHASTIQUE

NOIRHOMME

|           |                     |                       |
|-----------|---------------------|-----------------------|
| 14 heures | BERNARD<br>MAREE    | JEAN-MARIE<br>JACQUES |
| 15 heures | HAZARD<br>DAHMAN    | PASCAL<br>GUY         |
| 16 heures | VAN GAEVER<br>CAPON | RUDY<br>FRANCOIS      |
| 17 heures | LIEPIN<br>WENZI     | SERGE<br>MENAYANE     |

COURS DE SYSTEMES D'EXPLOITATIONS

RAMAEKERS

|           |                               |                                       |
|-----------|-------------------------------|---------------------------------------|
| 14 heures | MULLER<br>WARZEE<br>GUILLAUME | JOSIANE<br>PIERRE<br>GILBERT          |
| 15 heures | HALLO<br>WATERKEYN<br>JOSIS   | MARIA<br>PASCAL<br>ALAIN              |
| 16 heures | DANGIS<br>DE SWERT<br>RYS     | JEAN-MICHEL<br>PIERRES-YVES<br>XAVIER |
| 17 heures | BOUKHCHA<br>DJNANT            | SAID<br>JEAN-MARC                     |

\*\*\*\*\*  
 \* LE 16 JUIN AU MATIN \*  
 \*\*\*\*\*

COURS DE THEORIE DES GRAPHS

FICHEFET

|              |                               |                                   |
|--------------|-------------------------------|-----------------------------------|
| 8 heures 30  | VAUSE<br>DEVAUX<br>DELAITE    | MICHEL<br>YVES<br>YVES            |
| 9 heures 30  | DEBOIS<br>MENDE<br>VAN DOORNE | J.-J.<br>KONGA<br>JEAN-LOUIS      |
| 10 heures 30 | HUCO<br>CLANTIN<br>DUBUISSON  | BENOIT<br>MARCEL<br>PIERRE        |
| 11 heures 30 | ELIAS<br>AUBERT<br>MORAUX     | CHRISTIAN<br>BERNARD<br>FRANCOISE |



## COURS DE PROCESSUS STOCHASTIQUE

NOIRHOMME

|              |                                        |                                    |
|--------------|----------------------------------------|------------------------------------|
| 8 heures 30  | WILLIQUET                              | MARC                               |
| 9 heures 30  | AGUILAR<br>BAUDOUX<br>PICHOT<br>MARTIN | VICTOR<br>GUY<br>MICHEL<br>VINCENT |
| 10 heures 30 | MOULART<br>TRIGEAUX                    | PASCAL<br>BENOIT                   |
| 11 heures 30 | HENDRICK<br>JACQUET                    | ANNE<br>JEAN-MARIE                 |

## COURS DE SYSTEMES D'EXPLOITATIONS

RAMAEKERS

|              |                                   |                               |
|--------------|-----------------------------------|-------------------------------|
| 8 heures 30  | VERRIEST<br>VANGEERSDAEL<br>GUIOT | ALAIN<br>JOELLE<br>DIDIER     |
| 9 heures 30  | TERRYN<br>TOURNAY<br>GUILLAUME    | PASCALE<br>PATRICK<br>GILBERT |
| 10 heures 30 | HOANG<br>VANHEMELRYCK<br>GILLIN   | HOA<br>CLAUDE<br>PATRICK      |
| 11 heures 30 | MOONS<br>PIERRE<br>VANIMPE        | XAVIER<br>BRUNO<br>M-FR       |

\*\*\*\*\*  
 \* LE 16 JUIN APRES-MIDI \*  
 \*\*\*\*\*

## COURS DE THEORIE DES GRAPHS

FICHEFET

|           |                                           |                                   |
|-----------|-------------------------------------------|-----------------------------------|
| 14 heures | BEYNE-MAIRIAUX<br>PIERRE<br>VANHENTENRYCK | FR.<br>BERNARD<br>PASCAL<br>BRUNO |
| 15 heures | JEANJOT<br>DEMOULIN<br>KAYSER             | JEAN-PIERRE<br>PASCAL<br>MANOEL   |
| 16 heures | VISART<br>MANISE<br>CAVEZ                 | PATRICK<br>J.J.<br>J.M.           |
| 17 heures | SOUILBIN<br>DHEUR<br>WEICKER              | EMMANUEL<br>MICHEL                |

## COURS DE PROCESSUS STOCHASTIQUE

NOIRHOMME

|           |                               |                                |
|-----------|-------------------------------|--------------------------------|
| 14 heures | DETAILLE<br>LEFEBVRE<br>GATHY | J.-CL<br>PASCALE<br>J.M.       |
| 15 heures | LE KHAM                       |                                |
| 16 heures | VLIETINCK<br>GPAAS<br>CARLIER | MICHEL<br>ROBERT<br>JEAN-LOUIS |
| 17 heures | LEROY<br>LABRANCHE            | ANDRE<br>MARC                  |



## COURS DE SYSTEMES D'EXPLOITATIONS

RAMAEKERS

|           |            |                |
|-----------|------------|----------------|
| 14 heures | JUSSERET   | MARI-CHRISTINE |
|           | HAZARD     | PASCAL         |
| 15 heures | DAHMEN     | GUY            |
|           | VAN GAEVER | RUDY           |
|           | CAPON      | FRANCOIS       |
| 16 heures | LIEPIN     | SERGE          |
|           | EVARD      | MARC           |
|           | WENZI      | MENAYANE       |
| 17 heures | ROTENS     | ETIENNE        |
|           | RIFFONT    | CHRISTINE      |
|           | DUCARME    | PHILIPPE       |
|           | VANDEPUT   | MICHEL         |

\*\*\*\*\*  
 \* LE 17 JUIN AU MATIN \*  
 \*\*\*\*\*

## COURS DE THEORIE DES GRAPHES

FICHEFET

|              |          |            |
|--------------|----------|------------|
| 8 heures 30  | OGER     | LAURA      |
|              | PIREN    | SYLVAIN    |
| 9 heures 30  | CADELLI  | MARIO      |
|              | BURNAY   | JEAN       |
|              | BOUCHAT  | PIERRE     |
| 10 heures 30 | MEURISSE | MYRIAM     |
|              | VALENTIN | DIDIER     |
|              | STAES    | OLIVIER    |
| 11 heures 30 | CANART   | JEAN-YVES  |
|              | BERNARD  | JEAN-MARIE |
|              | MAREE    | JACQUES    |

## COURS DE PROCESSUS STOCHASTIQUE

NOIRHOMME

|              |           |           |
|--------------|-----------|-----------|
| 8 heures 30  | RENAULD   | YVAN      |
|              | VAUSE     | MICHEL    |
|              | DEVAUX    | YVES      |
| 9 heures 30  | DEBOIS    | J.-J.     |
|              | MENDE     | KONGA     |
| 10 heures 30 | CLANTIN   | MARCEL    |
|              | DUBUISSON | PIERRE    |
| 11 heures 30 | ELIAS     | CHRISTIAN |

MORAU

FRANCOISE

## COURS DE SYSTEMES D'EXPLOITATIONS

RAMAEKERS

|              |           |            |
|--------------|-----------|------------|
| 8 heures 30  | WILLIQUET | MARC       |
|              | AGUILAR   | VICTOR     |
| 9 heures 30  | BAUDOUX   | GUY        |
|              | PICHOT    | MICHEL     |
|              | MARTIN    | VINCENT    |
| 10 heures 30 | MOULART   | PASCAL     |
|              | TRIGEAUX  | BENOIT     |
|              | WILMES    | HORST      |
| 11 heures 30 | HENDRICK  | ANNE       |
|              | JACQUET   | JEAN-MARIE |
|              | LORGE     | FREDDY     |
|              | MULLER    | DENISE     |



\*\*\*\*\*  
 \* LE 17 JUIN APRES-MIDI \*  
 \*\*\*\*\*

# COURS DE THEORIE DES GRAPHS

FICHEFET

|           |                                   |                               |
|-----------|-----------------------------------|-------------------------------|
| 14 heures | VERRIEST<br>VANGEERSDAEL<br>GUIOT | ALAIN<br>JOELLE<br>DIDIER     |
| 15 heures | TERRYN<br>TOURNAY<br>GUILLAUME    | PASCALE<br>PATRICK<br>GILBERT |
| 16 heures | HOANG<br>VANHEMELRYCK<br>GILLIN   | HOA<br>CLAUDE<br>PATRICK      |
| 17 heures | MOONS<br>PIERRE<br>VANINPE        | XAVIER<br>BRUNO<br>M-FR       |

# COURS DE PROCESSUS STOCHASTIQUE

NOIRHOMME

|           |                    |                        |
|-----------|--------------------|------------------------|
| 14 heures | MULLER<br>WARZEE   | JOSIANE<br>PIERRE      |
| 15 heures | GUILLAUME<br>HALLO | GILBERT<br>MARIA       |
| 16 heures | JOSIS<br>DANGIS    | ALAIN<br>JEAN-MICHEL   |
| 17 heures | DE SWERT<br>RYS    | PIERRES-YVES<br>XAVIER |

# COURS DE SYSTEMES D'EXPLOITATIONS

RAMAEKERS

|           |                                           |                                |
|-----------|-------------------------------------------|--------------------------------|
| 14 heures | BEYNE-MAIRIAUX<br>PIERRE<br>VANHENTENRYCK | FR.<br>BERNARD<br>PASCAL       |
| 15 heures | JEANJOT<br>DEMOULIN<br>KAYSER             | BRUNO<br>JEAN-PIERRE<br>PASCAL |
| 16 heures | VISART<br>MANISE<br>CAVEZ                 | MANOEL<br>PATRICK<br>J.J.      |
| 17 heures | SQUILBIN<br>DHEUR<br>WEICKER              | J.M.<br>EMMANUEL<br>MICHEL     |

\*\*\*\*\*  
 \* LE 20 JUIN AU MATIN \*  
 \*\*\*\*\*

# COURS DE THEORIE DES GRAPHS

FICHEFET

|              |                                |                                  |
|--------------|--------------------------------|----------------------------------|
| 8 heures 30  | HAZARD<br>DAHMEN<br>VAN GAEVER | PASCAL<br>GUY<br>RUDY            |
| 9 heures 30  | CAPON<br>LIEPIN<br>EVRARD      | FRANCOIS<br>SERGE<br>MARC        |
| 10 heures 30 | WENZI<br>ROTENS<br>RIFFONT     | MENAYANE<br>ETIENNE<br>CHRISTINE |
| 11 heures 30 | DUCHARME<br>VANDEPUT           | PHILIPPE<br>MICHEL               |



## COURS DE PROCESSUS STOCHASTIQUE

NOIRHOMME

|              |                            |                          |
|--------------|----------------------------|--------------------------|
| 8 heures 30  | CAVEZ<br>SQUILBIN<br>DHEUR | J.J.<br>J.M.<br>EMMANUEL |
| 9 heures 30  | WEICKER<br>OGER            | MICHEL<br>LAURA          |
| 10 heures 30 | PIREN<br>CADELLI           | SYLVAIN<br>MARIO         |
| 11 heures 30 | BURNAY<br>BOUCHAT          | JEAN<br>PIERRE           |

## COURS DE THEORIE DES ORGANISATIONS

DRABS

|              |                                                                               |                                                              |
|--------------|-------------------------------------------------------------------------------|--------------------------------------------------------------|
| 8 heures 30  | HUCQ                                                                          | BENOIT                                                       |
| 9 heures 30  | CLANTIN<br>DUBUISSON<br>ELIAS<br>AUBERT<br>MORAUX<br>BEYNE-MAIRIAUX<br>PIERRE | MARCEL<br>PIERRE<br>CHRISTIAN<br>BERNARD<br>FRANCOISE<br>FR. |
| 10 heures 30 | VANHENTENRYCK<br>JEANJOT<br>DEMOULIN                                          | BERNARD<br>PASCAL<br>BRUNO                                   |
| 11 heures 30 | KAYSER<br>VISART<br>MANISE                                                    | JEAN-PIERRE<br>PASCAL<br>MANOEL<br>PATRICK                   |

## COURS DE SYSTEMES D'EXPLOITATIONS

RAMAEKERS

|              |                                 |                                     |
|--------------|---------------------------------|-------------------------------------|
| 8 heures 30  | DETAILLE<br>LEFEBVRE<br>GATHY   | J.-CL<br>PASCALE<br>J.M.            |
| 9 heures 30  | LE KHAM<br>VLIETINCK<br>LEBEGUE | MICHEL<br>HENRY                     |
| 10 heures 30 | TASIAUX<br>GRAAS<br>CARLIER     | BRIGITTE<br>ROBERT                  |
| 11 heures 30 | LEROY<br>LABRANCHE<br>RENAULD   | JEAN-LOUIS<br>ANDRE<br>MARC<br>YVAN |

## COURS DE INFORMATIQUE ET SCIENCES

BERLEUR

|              |                                         |                                     |
|--------------|-----------------------------------------|-------------------------------------|
| 8 heures 30  | WILLIQUET<br>MARTIN                     | MARC<br>VINCENT                     |
| 9 heures 30  | MOULART<br>TRIGEAUX<br>WILMES<br>MULLER | PASCAL<br>BENOIT<br>HORST<br>DENISE |
| 10 heures 30 | VERRIEST<br>VANGEERSDAEL                | ALAIN<br>JOELLE                     |
| 11 heures 30 | HOANG<br>VANHAMELRYCK                   | HOA<br>CLAUDE                       |



\*\*\*\*\*  
 \* LE 20 JUIN APRES-MIDI \*  
 \*\*\*\*\*

COURS DE THEORIE DES GRAPHS

FICHEFET

|           |                               |                                       |
|-----------|-------------------------------|---------------------------------------|
| 14 heures | MULLER<br>WARZEE<br>GUILLAUME | JOSIANE<br>PIERRE<br>GILBERT          |
| 15 heures | HALLO<br>WATERKEYN<br>JOSIS   | MARIA<br>PASCAL<br>ALAIN              |
| 16 heures | DANGIS<br>DE SWERT<br>RYS     | JEAN-MICHEL<br>PIERRES-YVES<br>XAVIER |
| 17 heures | BOUKHCHA<br>DINANT            | SAID<br>JEAN-MARC                     |

\*\*\*\*\*  
 \* LE 21 JUIN AU MATIN \*  
 \*\*\*\*\*

COURS DE PASCAL

CARDINAEL

EPREUVE ECRITE

COURS DE SYSTEMES D'EXPLOITATIONS

RAMAEKERS

|              |                               |                                |
|--------------|-------------------------------|--------------------------------|
| 8 heures 30  | OGER<br>PIREN<br>CADELLI      | LAURA<br>SYLVAIN<br>MARIO      |
| 9 heures 30  | BURNAY<br>BOUCHAT<br>MEURISSE | JEAN<br>PIERRE<br>MYRIAM       |
| 10 heures 30 | VALENTIN<br>STAES<br>CANART   | DIDIER<br>OLIVIER<br>JEAN-YVES |
| 11 heures 30 | BERNARD<br>MAREE              | JEAN-MARIE<br>JACQUES          |

DE PAUW

FABIENNE

COURS DE INFORMATIQUE ET SCIENCES

BERLEUR

|              |                               |                                 |
|--------------|-------------------------------|---------------------------------|
| 8 heures 30  | CLANTIN<br>DUBUISSON<br>ELIAS | MARCEL<br>PIERRE<br>CHRISTIAN   |
| 9 heures 30  | AUBERT<br>MORAUX<br>PIERRE    | BERNARD<br>FRANCOISE<br>BERNARD |
| 10 heures 30 | VANHENTENRYCK<br>JEANJOT      | PASCAL<br>BRUNO                 |
| 11 heures 30 | DEMOULIN<br>KAYSER            | JEAN-PIERRE<br>PASCAL           |



\*\*\*\*\*  
 \* LE 21 JUIN APRES-MIDI \*  
 \*\*\*\*\*

COURS DE PASCAL

CARDINAEL

|           |                                                                                                                                                                                                                                                                                                                                              |                                                                                                                                                                                                                                                                                                     |
|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 14 heures | WILLIQUET<br>AGUILAR<br>BAUDOUX<br>PICHOT<br>WILMES<br>HENDRICK<br>JACQUET<br>LORGE<br>GUIOT<br>TERRYN<br>TOURNAY<br>GUILLAUME<br>GILLIN<br>PIERRE<br>VANIMPE<br>DETAILLE<br>LEFEBVRE<br>GATHY<br>LE KHAM<br>VLIETINCK<br>LEBEGUE<br>CARLIER<br>LEROY<br>LABRANCHE<br>RENAULD<br>VAUSE<br>DEVAUX<br>DELAITE<br>DEBOIS<br>MENDE<br>VAN DOORNE | MARC<br>VICTOR<br>GUY<br>MICHEL<br>HORST<br>ANNE<br>JEAN-MARIE<br>FREDDY<br>DIDIER<br>PASCALE<br>PATRICK<br>GILBERT<br>PATRICK<br>BRUNO<br>M-FR<br>J.-CL<br>PASCALE<br>J.M.<br><br>MICHEL<br>HENRY<br>JEAN-LOUIS<br>ANDRE<br>MARC<br>YVAN<br>MICHEL<br>YVES<br>YVES<br>J.-J.<br>KONGA<br>JEAN-LOUIS |
| 15 heures |                                                                                                                                                                                                                                                                                                                                              |                                                                                                                                                                                                                                                                                                     |
| 16 heures |                                                                                                                                                                                                                                                                                                                                              |                                                                                                                                                                                                                                                                                                     |

|           |                                                                                                             |                                                                                                      |
|-----------|-------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|
| 17 heures | BEYNE-MAIRIAUX<br>VISART<br>MANISE<br>CAVEZ<br>DHEUR<br>WEICKER<br>ROTENS<br>RIFFONT<br>DUCARME<br>VANDEPUT | FR.<br>MANOEL<br>PATRICK<br>J.J.<br>EMMANUEL<br>MICHEL<br>ETIENNE<br>CHRISTINE<br>PHILIPPE<br>MICHEL |
|-----------|-------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|

\*\*\*\*\*  
 \* LE 22 JUIN AU MATIN \*  
 \*\*\*\*\*

COURS DE INFORMATIQUE ET SCIENCES

BERLEUR

|              |                               |                                      |
|--------------|-------------------------------|--------------------------------------|
| 8 heures 30  | DE PAUW<br>JUSSERET<br>HAZARD | FABIENNE<br>MARI-CHRISTINE<br>PASCAL |
| 9 heures 30  | DAHMEN<br>VAN GAEVER          | GUY<br>RUDY                          |
| 10 heures 30 | CAPON<br>LIEPIN               | FRANCOIS<br>SERGE                    |
| 11 heures 30 | EVARD<br>WENZI                | MARC<br>MENAYANE                     |

\*\*\*\*\*  
 \* LE 23 JUIN AU MATIN \*  
 \*\*\*\*\*

COURS DE THEORIE DES LANGAGES

LEROY

EPREUVE ECRITE



\*\*\*\*\*  
\* LE 24 JUIN AU MATIN \*  
\*\*\*\*\*

COURS DE COBOL

VAN LAMSWEERDE

EPREUVE ECRITE

\*\*\*\*\*  
\* LE 27 JUIN AU MATIN \*  
\*\*\*\*\*

COURS DE PROBABILITE ET STATISTIQUE

LOUVEAUX

EPREUVE ECRITE

\*\*\*\*\*  
\* LE 27 JUIN APRES-MIDI \*  
\*\*\*\*\*

COURS DE PROBABILITE ET STATISTIQUE

LOUVEAUX

14 heures

WILLIQUET  
AGUILAR  
BAUDOUX  
PICHOT  
MARTIN  
MOULART  
TRIGEAUX  
LORGE

MARC  
VICTOR  
GUY  
MICHEL  
VINCENT  
PASCAL  
BENOIT  
FREDDY  
DENISE

15 heures

MULLER  
VERRIEST  
VANGEERSDAEL  
MOONS  
DETAILLE  
LEFEBVRE  
GATHY  
LE KHAM

ALAIN  
JOELLE  
XAVIER  
J.-CL  
PASCALE  
J.M.

16 heures

VLIETINCK  
LEBEGUE  
TASIAUX  
CARLIER  
LEROY  
LABRANCHE  
RENAULD  
VAUSE  
DELAITE  
MORAUX  
BEYNE-MAIRIAUX  
PIERRE  
VISART

MICHEL  
HENRY  
BRIGITTE  
JEAN-LOUIS  
ANDRE  
MARC  
YVAN  
MICHEL  
YVES  
FRANCOISE  
FR.

17 heures

SQUILBIN  
OGER  
PIREN  
MEURISSE  
VALENTIN  
BERNARD  
DE PAUW  
MULLER  
GUILLAUME

BERNARD  
MANOEL  
J.M.  
LAURA  
SYLVAIN  
MYRIAM  
DIDIER  
JEAN-MARIE  
FABIENNE  
JOSIANE  
GILBERT

\*\*\*\*\*

\* LE 29 JUIN AU MATIN \*

COURS DE CONCEPTION DE FICHIERS ET B.D.  
EPREUVE ECRITE

HAINAUT

\*\*\*\*\*  
\* LE 30 JUIN AU MATIN \*

COURS DE METHODO DE LA PROGRAMMATION  
EPREUVE ECRITE

LEROY



B I B L I O G R A P H I E

# BIBLIOGRAPHIE

- Broder S. "Final examination scheduling"  
Comm.Assoc. Com. Mach. Vol 7, p.494 (1964)
- Cole, A.J. "The preparation of examination time-tables using  
a small-store computer"  
The computer Journal Vol 7, p 117 (1964)
- Heller, Nelson B. "Mathematical methods of examination scheduling"  
A thesis submitted in partial fulfillment  
of the requirements for the degree of master  
of science, M.I.T. (1963)
- Kirchgassner, klaus, "Some algorithms for solving parts of  
the class scheduling problem".  
Internal Res. Rep, M.I.T. (Janvier 1963)
- Kirchgassner, Klaus, "Some aspects of class schedule assignment  
problems (graph theoretical approach)" Internal Res.Rep.MIT (1963)
- Oakford, R.V. Allen, D.W. and Chatterton, Lynne A."school  
scheduling - practice and theory".  
J; educational. data. Processing IV,1  
(67)
- Carlson, R.C. and Nemhauser, G.L. "Scheduling to minimize  
interaction cost" Operations Res.14  
(1966)
- Appleby, J.S.Blake, D.V. and Newman, E.A.(1961)  
"Techniques for producing school time-  
tables on a computer and their appli-  
cations to other scheduling problems"  
The computer journal Vol 3 p.237 (1961)
- Csima, J and Gotlieb C.C. "A Computer Method for construc-  
ting school timetables"  
Presented at the Eighteenth annual  
conference of the association for  
computing machinery, Denver, Colora-  
do (1963)
- Sherman, G.R. "A Combinatorial Problem Arising from scheduling  
of University classes" Journal of the Tennessee  
Academy of Science, Vol 38, N°3, p.115 (1963)



Oswald Henkes "Etablissement d'horaires de cours par coloration de graphes". Mémoire présenté en vue de l'obtention du grade de maître en informatique. F.N.D.P. Namur (1982)